



**TUGAS AKHIR - TE 145561**

**RANCANG BANGUN LENGAN ROBOT PENJEPIT PCB 3 DOF  
BERBASIS ARDUINO UNTUK PROSES *ETCHING* PCB  
OTOMATIS**

Gita Rizka Anandya  
NRP 2214030053

Dosen Pembimbing  
Fajar Budiman, ST., MSc.

PROGRAM STUDI KOMPUTER KONTROL  
Departemen Teknik Elektro Otomasi  
Fakultas Vokasi  
Institut Teknologi Sepuluh Nopember  
Surabaya 2017





***FINAL PROJECT - TE 145561***

***DESIGN OF 3 DOF PCB GRIPPER ROBOTIC ARM BASED ON  
ARDUINO FOR AUTOMATIC PCB ETCHING PROCESS***

Gita Rizka Anandya  
NRP 2214030053

*Advisor*  
Fajar Budiman, ST., MSc.

***COMPUTER CONTROL STUDY PROGRAM  
Electrical and Automation Engineering Department  
Vocational Faculty  
Institut Teknologi Sepuluh Nopember  
Surabaya 2017***



## PERNYATAAN KEASLIAN TUGAS AKHIR

Dengan ini saya menyatakan bahwa isi sebagian maupun keseluruhan Tugas Akhir saya dengan judul “**Rancang Bangun Lengan Robot Penjepit PCB 3 DOF Berbasis Arduino Untuk Proses *Etching* PCB Otomatis**” adalah benar-benar hasil karya intelektual mandiri, diselesaikan tanpa menggunakan bahan-bahan yang tidak diijinkan dan bukan merupakan karya pihak lain yang saya akui sebagai karya sendiri.

Semua referensi yang dikutip maupun dirujuk telah ditulis secara lengkap pada daftar pustaka.

Apabila ternyata pernyataan ini tidak benar, saya bersedia menerima sanksi sesuai peraturan yang berlaku.

Surabaya, 18 Juli 2017



Gita Rizka Anandya  
NRP 2214030053

-----Halaman ini sengaja dikosongkan-----


**RANCANG BANGUN LENGAN ROBOT PENJEPIT PCB 3 DOF  
BERBASIS ARDUINO UNTUK PROSES *ETCHING* PCB  
OTOMATIS**

**TUGAS AKHIR**

Diajukan Guna Memenuhi Sebagian Persyaratan  
Untuk Memperoleh Gelar Ahli Madya  
Pada  
Program Studi Komputer Kontrol  
Departemen Teknik Elektro Otomasi  
Fakultas Vokasi  
Institut Teknologi Sepuluh Nopember

Menyetujui:

Dosen Pembimbing

  
Fajar Budiman ST., MSc.

NIP. 19860707 201404 1 001

**SURABAYA  
JULI, 2017**

-----Halaman ini sengaja dikosongkan-----



# **RANCANG BANGUN LENGAN ROBOT PENJEPIT PCB 3 DOF BERBASIS ARDUINO UNTUK PROSES *ETCHING* PCB OTOMATIS**

**Nama : Gita Rizka Anandya**  
**Pembimbing : Fajar Budiman ST., MSc.**

## **ABSTRAK**

Saat ini banyak orang yang menggunakan PCB polos untuk membuat rangkaian karena dengan menggunakan PCB polos kita dapat membuat rangkaian yang membutuhkan banyak jalur tanpa harus menggunakan banyak kabel *jumper*. Untuk menggunakan PCB polos terdapat beberapa proses yang harus dilewati, mulai dari penyablonan, *etching* serta pengeboran. Dari ketiga proses tersebut, proses yang membutuhkan banyak waktu ialah *etching* karena menunggu tembaga pada PCB larut dalam campuran larutan  $HCL$ ,  $H_2O_2$  dan  $H_2O$ . kemudian harus di bilas dengan air bersih untuk menghilangkan bekas larutan  $HCL$ ,  $H_2O_2$  dan  $H_2O$  tersebut.

Oleh karena itu diperlukan sebuah alat untuk membuat proses *etching* tersebut berjalan secara otomatis. Alat tersebut bekerja dengan cara membawa PCB ke dalam campuran larutan  $HCL$ ,  $H_2O_2$  dan  $H_2O$  lalu menggerak-gerakkannya untuk mempercepat proses pelarutan tembaga. Setelah tembaga larut, maka secara otomatis alat akan membawa PCB ke dalam air bersih untuk menghilangkan sisa campuran larutan  $HCL$ ,  $H_2O_2$  dan  $H_2O$ . Alat ini terdiri dari sebuah lengan robot 3 dof dengan sistem *inverse kinematic* dan objek *gripper* untuk penjepit PCB.

Hasil dari Tugas Akhir ini adalah lengan robot yang dapat melakukan proses *etching*. Berdasarkan pengujian *output* servo didapatkan masing-masing motor servo memiliki kesalahan dalam menghasilkan sudut, berdasarkan pengujian daya cengkram *gripper* batas ukuran PCB maksimal 8x4, dan berdasarkan pengujian proses *etching* hanya memerlukan waktu sekitar 1 menit 30 detik. Serta berdasarkan pengujian keseluruhan sistem, terdapat kegagalan pada pemindahan PCB.

**Kata Kunci :** PCB, *Etching*,  $HCL$ ,  $H_2O_2$  dan  $H_2O$ , Lengan Robot, *Inverse Kinematic*

-----Halaman ini sengaja dikosongkan-----

## **DESIGN OF 3 DOF PCB GRIPPER ROBOTIC ARM BASED ON ARDUINO FOR AUTOMATIC PCB ETCHING PROCESS**

**Name : Gita Rizka Anandya**  
**Advisor : Fajar Budiman ST., MSc.**

### **ABSTRACT**

*Currently many people are using plain PCB to make the circuit because by using plain PCB we can make a circuit that takes many paths without having to use many jumper cables. To use a plain PCB there are several processes that must be passed, ranging from penyablonan, etching and drilling. Of the three processes, the process that takes a lot of time is etching because waiting for copper on the PCB dissolves in the liquid HCL,  $H_2O_2$  and  $H_2O$  then must be rinsed with clean water to remove the former HCL,  $H_2O_2$  and  $H_2O$  solution.*

*Therefore a tool is needed to make the etching process run automatically. The device works by bringing the PCB into the HCL,  $H_2O_2$  and  $H_2O$  solution and then moving it to speed up the copper dissolution process. After the copper dissolves, it will automatically bring the PCB into clean water to remove the residual HCL,  $H_2O_2$  and  $H_2O$  solution. This tool consists of a 3 dof robot arm with a kinematic inverse system and a gripper object for PCB clamps.*

*The result of this final project is a robot arm that can do the etching process. Based on the servo output test, each servo motor has an error in producing angle, based on gripper gripper puncture test of maximal 8x4 PCB size, and based on testing etching process only takes about 1 minute and 30 seconds. As well as on the whole system test, there is a failure on PCB removal.*

**Keywords :** *PCB, Etching, HCL,  $H_2O_2$  and  $H_2O$ , Robotic arm, Inverse Kinematic*

-----Halaman ini sengaja dikosongkan-----

## KATA PENGANTAR

Puji syukur penulis panjatkan kehadiran Allah SWT yang selalu memberikan rahmat dan hidayah-Nya sehingga Tugas Akhir ini dapat terselesaikan dengan baik. Shalawat serta salam semoga selalu dilimpahkan kepada Rasulullah Muhammad SAW, keluarga, sahabat, dan umat muslim yang senantiasa meneladani beliau.

Tugas Akhir ini disusun untuk memenuhi sebagian persyaratan guna menyelesaikan pendidikan Diploma-3 pada Program Studi Komputer Kontrol, Departemen Teknik Elektro Otomasi, Fakultas Vokasi, Institut Teknologi Sepuluh Nopember Surabaya dengan judul:

### **RANCANG BANGUN LENGAN ROBOT PENJEPIT PCB 3 DOF BERBASIS ARDUINO UNTUK PROSES *ETCHING* OTOMATIS**

Dalam Tugas Akhir ini, rancang bangun lengan robot penjepit pcb 3 DOF berbasis Arduino yang digunakan untuk memindahkan PCB dari cairan NaOH ke campuran larutan  $HCL$ ,  $H_2O_2$  dan  $H_2O$  dan di goyang-goyangkan untuk melarutkan tembaga kemudian dipindahkan lagi ke air bersih untuk membilas sisa campuran larutan  $HCL$ ,  $H_2O_2$  dan  $H_2O$ .

Penulis mengucapkan terima kasih kepada Ibu dan Bapak penulis yang memberikan berbagai bentuk doa serta dukungan tulus tiada henti, Bapak Fajar Budiman ST., M.T. atas segala bimbingan ilmu, moral, dan spiritual dari awal hingga terselesaikannya Tugas Akhir ini, keluarga, sahabat, dan teman yang selalu memberikan doa, semangat, dan dukungannya kepada penulis. Penulis juga mengucapkan banyak terima kasih kepada semua pihak yang telah membantu baik secara langsung maupun tidak langsung dalam proses penyelesaian Tugas Akhir ini.

Penulis menyadari dan memohon maaf atas segala kekurangan pada Tugas Akhir ini. Akhir kata, semoga Tugas Akhir ini dapat bermanfaat dalam pengembangan keilmuan di kemudian hari.

Surabaya, 18 Juli 2017

Penulis

-----Halaman ini sengaja dikosongkan-----

# DAFTAR ISI

## HALAMAN

HALAMAN JUDUL .....	i
HALAMAN JUDUL .....	i
PERNYATAAN KEASLIAN TUGAS AKHIR .....	v
HALAMAN PENGESAHAN .....	vii
ABSTRAK .....	ix
<i>ABSTRACT</i> .....	xi
KATA PENGANTAR .....	xiii
DAFTAR ISI .....	xv
DAFTAR GAMBAR .....	xix
DAFTAR TABEL .....	xxi
1 BAB I PENDAHULUAN .....	1
1.1 Latar Belakang .....	1
1.2 Permasalahan .....	1
1.3 Tujuan .....	2
1.4 Batasan Masalah .....	2
1.5 Metodologi Penelitian .....	2
1.6 Sistematika Laporan .....	3
1.7 Relevansi .....	3
2 BAB II TEORI DASAR .....	5
2.1 Lengan Robot .....	5
2.1.1 Konsep Dasar Lengan Robot .....	6
2.1.2 Bagian-Bagian Lengan Robot .....	7
2.2 Motor Servo .....	7
2.2.1 Konstruksi Motor Servo .....	9
2.2.2 Prinsip Kerja Motor Servo .....	11
2.3 <i>Switch-Mode Power Supply</i> .....	12
2.4 Arduino Nano .....	13
2.4.1 Perangkat Keras Arduino .....	13
2.4.2 Spesifikasi Arduino Nano .....	14
2.4.3 IDE ( <i>Integrated Development Environment</i> ) .....	15
2.4.4 Sinyal PWM pada Arduino .....	18
2.5 <i>Buck DC to DC Converter</i> .....	19

2.6	<i>Degrees of Freedom</i> .....	20
2.7	Kinematika Robot .....	21
2.7.1	Kinematik Robot Lengan Dua Sendi .....	21
2.7.2	Kinematik Robot Lengan Tiga Sendi.....	23
2.8	Larutan H3 Pengganti FeCl3 untuk <i>Etching</i> PCB.....	25
2.9	<i>Slider</i> sebagai Penggerak Lengan Robot .....	26
3	BAB III PERANCANGAN SISTEM KONTROL .....	29
3.1	Blok Fungsional Sistem .....	29
3.2	Pembuatan Perangkat Elektronik .....	31
3.2.1	Pengkabelan Arduino Nano Dengan Motor Servo.....	31
3.2.2	Rangkaian Komunikasi Data Serial Antara Arduino .....	32
3.3	Pembuatan Perangkat Mekanik.....	33
3.3.1	Pembuatan Lengan Robot .....	33
3.4	Pembuatan Perangkat Lunak.....	36
3.4.1	Perhitungan <i>Inverse Kinematic</i> Lengan Robot.....	37
3.4.2	Pembuatan <i>Flowchart Program</i> .....	39
3.4.3	Pendefinisian Pin dan Pustaka .....	41
3.4.4	Segmen Program Untuk Pengaturan Awal.....	41
3.4.5	Segmen Program Untuk Menggerakkan Servo .....	42
3.4.6	Setting Program Komunikasi Serial Arduino.....	42
4	BAB IV HASIL SIMULASI DAN IMPLEMENTASI .....	47
4.1	Data Pengukuran <i>Switch-Mode Power Supply</i> .....	47
4.2	Data Pengukuran <i>Buck Converter DC to DC</i> .....	47
4.3	Pengukuran Dan Pengujian Arduino Nano Dengan Motor Servo .. .....	48
4.4	Pengujian Daya Cengkram <i>Gripper</i> .....	50
4.5	Pengukuran Kecepatan Putaran <i>Gripper</i> .....	51
4.6	Pengukuran Dan Pengujian Lama Proses <i>Etching</i> Pada Berbagai Ukuran PCB.....	52
4.7	Pengukuran dan Pengujian Kecepatan Putaran PCB Terhadap Lama Pelarutan Tembaga .....	53
4.8	Pengujian Keseluruhan Sistem .....	54
5	BAB V PENUTUP .....	59
5.1	Kesimpulan .....	59
5.2	Saran .....	59
6	DAFTAR PUSTAKA .....	61



LAMPIRAN A .....	63
A.1 Lampiran Program Arduino .....	63
LAMPIRAN B.....	71
B.1 Dokumentasi .....	71
LAMPIRAN C.....	73
C.1 <i>Datasheet</i> Motor Servo MG995.....	73
C.2 <i>Datasheet</i> Motor Servo MG90.....	75
C.3 <i>Datasheet Buck DC To DC Converter</i> XL4005 .....	77
LAMPIRAN D .....	87
DAFTAR RIWAYAT HIDUP .....	87

-----Halaman ini sengaja dikosongkan-----

## DAFTAR GAMBAR

### HALAMAN

Gambar 2.1	Lengan Robot 3 Dof.....	5
Gambar 2.2	Bentuk Fisik Motor Servo MG995 .....	8
Gambar 2.3	Bentuk Fisik Motor Servo SG90.....	8
Gambar 2.4	<i>Pin Out</i> Kabel Motor Servo .....	9
Gambar 2.5	Pulsa Kendali Motor Servo .....	10
Gambar 2.6	Arah Putaran Motor Servo <i>Standard</i> .....	11
Gambar 2.7	Gambar Rangkaian <i>Switch Mode Power Supply</i> .....	12
Gambar 2.8	Bagian – Bagian Arduino Nano .....	13
Gambar 2.9	Tampilan Awal IDE Arduino.....	15
Gambar 2.10	Tampilan Arduino Ketika Tidak Terjadi <i>Error</i> Pada Saat <i>Verify Program</i> .....	17
Gambar 2.11	Memilih <i>Board</i> Arduino Pada Arduino .....	17
Gambar 2.12	Memilih <i>Port</i> Pada Arduino.....	18
Gambar 2.13	Sinyal PWM dengan Nilai yang Diberikan.....	19
Gambar 2.14	Rangkaian Konverter DC-DC Tipe <i>Buck</i> .....	20
Gambar 2.15	Ilustrasi Lengan Robot 3 Sendi .....	20
Gambar 2.16	Konfigurasi Robot 2 Sendi (2 DOF) .....	21
Gambar 2.17	Konfigurasi Robot Tiga Sendi (3 DOF) .....	24
Gambar 2.18	Desain <i>Slider</i> Penggerak Lengan Robot .....	27
Gambar 3.1	Blok Diagram Fungsional Keseluruhan Sistem .....	29
Gambar 3.2	Blok Diagram Fungsional Lengan Robot .....	30
Gambar 3.3	Pengkabelan Arduino Nano Dengan <i>Motor Servo</i> .....	32
Gambar 3.4	Rangkaian Komunikasi Data Antar Arduino .....	33
Gambar 3.5	Rancangan Bagian Dasar Lengan Robot.....	34
Gambar 3.6	Rancangan Bagian Lengan Bawah Robot.....	34
Gambar 3.7	Rancangan Bagian Lengan Atas Robot.....	35
Gambar 3.8	Bagian Peralatan Lengan Robot.....	35
Gambar 3.9	Hasil Rancangan Keseluruhan Bagian .....	36
Gambar 3.10	Pemodelan Kinematik Lengan Robot .....	37
Gambar 3.11	Mencari Panjang $a$ .....	37
Gambar 3.12	Mencari Panjang $b$ .....	38
Gambar 3.13	Mencari Nilai Sudut $\beta$ .....	38
Gambar 3.14	Mencari Nilai Sudut $\theta$ Dan $\lambda$ .....	38

Gambar 3.15	<i>Flowchart</i> Keseluruhan Program.....	40
Gambar 3.16	Program Definisi Pustaka Dan Tipe Data.....	41
Gambar 3.17	Segmen Program Untuk Pengaturan Awal .....	41
Gambar 3.18	Segmen Program Untuk Menggerakkan Servo .....	42
Gambar 3.19	Program <i>Transfer</i> Komunikasi Serial .....	43
Gambar 3.20	<i>Flowchart</i> Pengiriman Data Komunikasi Serial .....	44
Gambar 3.21	Program <i>Receiver</i> Komunikasi Serial.....	45
Gambar 3.22	<i>Flowchart</i> Penerimaan Data Komunikasi Serial.....	46
Gambar 4.1	Grafik Data Pengukuran Output Servo .....	49
Gambar 4.2	Grafik Presentase Kesalahan Sudut Servo .....	50
Gambar 4.3	Pengujian Daya Cengkram <i>Gripper</i> .....	51

## DAFTAR TABEL

	HALAMAN
Tabel 2.1	Spesifikasi Motor Servo MG995 ..... 9
Tabel 2.2	Spesifikasi Motor Servo SG90..... 9
Tabel 2.3	Nilai Penyisalan Motor Servo ..... 10
Tabel 2.4	Fungsi Konfigurasi Arduino Nano..... 14
Tabel 2.5	Spesifikasi Arduino Nano ..... 14
Tabel 4.1	Data Pengukuran <i>Power Suplly</i> ..... 47
Tabel 4.2	Data Pengukuran <i>Buck Converter</i> ..... 47
Tabel 4.3	Data Pengukuran <i>Output</i> Servo..... 48
Tabel 4.4	Persentase Kesalahan Sudut Servo ..... 49
Tabel 4.5	Data Pengamatan Daya Cengkram <i>Gripper</i> ..... 50
Tabel 4.6	Data Pengamatan Kecepatan Putaran <i>Gripper</i> ..... 52
Tabel 4.7	Lama Proses <i>Etching</i> Pada Berbagai Ukuran PCB .... 52
Tabel 4.8	Kecepatan Putaran PCB Terhadap Lama Pelarutan Tembaga ..... 53
Tabel 4.9	Hasil Pengujian Dengan PCB Ukuran 3 x 5 ..... 54
Tabel 4.10	Hasil Pengujian Dengan PCB Ukuran 5 x 4 ..... 54
Tabel 4.11	Hasil Pengujian Dengan PCB Ukuran 5 x 6 ..... 55
Tabel 4.12	Hasil Pengujian Dengan PCB Ukuran 7 x 4 ..... 55
Tabel 4.13	Hasil Pengujian Dengan PCB Ukuran 8 x 4 ..... 55
Tabel 4.14	Hasil Pengujian Dengan PCB Ukuran 4 x 9 ..... 55
Tabel 4.15	Hasil Pengujian Dengan PCB Ukuran 5 x 9 ..... 56
Tabel 4.16	Hasil Pengujian Dengan PCB Ukuran 6 x 7 ..... 56
Tabel 4.17	Hasil Pengujian Dengan PCB Ukuran 7 X 7 ..... 56
Tabel 4.18	Hasil Pengujian Dengan PCB Ukuran 6 x 10 ..... 56

-----Halaman ini sengaja dikosongkan-----

# BAB I

## PENDAHULUAN

### 1.1 Latar Belakang

Bagi para pelaku industri manufaktur di Indonesia yang terus berkembang saat ini, kemampuan untuk beroperasi lebih cepat, lebih hemat biaya, dan lebih inovatif telah menjadi hal utama. Berdasarkan studi terbaru dari UNIDO (*United Nations Industrial Development Organization* atau Organisasi Pengembangan Industri PBB), kini, Indonesia adalah salah satu dari 10 negara manufaktur berpengaruh di dunia. Selain itu, semakin besar pula potensi Indonesia untuk meningkatkan *volume* dan kualitas eksportnya dengan mengadopsi revolusi industri 4.0.<sup>[1]</sup>

PCB *Etching Machine* merupakan alat yang digunakan untuk melarutkan bagian logam yang tidak dibutuhkan pada papapn PCB kosong menggunakan campuran larutan  $HCL$ ,  $H_2O_2$  dan  $H_2O$ . Untuk membuat proses *etching* tersebut berjalan secara otomatis, diperlukan sebuah lengan robot. Lengan robot tersebut bekerja dengan cara membawa PCB ke dalam larutan  $HCL$ ,  $H_2O_2$  dan  $H_2O$  lalu menggerak-gerakkannya untuk mempercepat proses pelarutan tembaga. Terdapat tiga wadah yang digunakan dalam proses *etching*, yang pertama adalah wadah yang berisi  $NaOH$ , yang kedua adalah  $HCL$ ,  $H_2O_2$  dan  $H_2O$  dan yang terakhir adalah air bersih. Lengan ini dapat melakukan *etching* secara otomatis dengan beberapa tumpukan PCB. Setelah tembaga larut, maka secara otomatis alat akan membawa PCB ke dalam air bersih untuk menghilangkan sisa larutan  $HCL$ ,  $H_2O_2$  dan  $H_2O$ .

Dalam pembuatan lengan robot ini diperlukan beberapa motor servo sebagai penggerak lengan, untuk menggerakkan motor servo tersebut diperlukan suatu modul servo *controller* untuk mengatur gerakan lengan robot agar dapat bergerak secara selaras dan seimbang. *Servo controller* sudah terdapat dalam motor servo itu sendiri. Untuk menggerakkan lengan robot digunakan suatu pengendali yang disebut mikrokontroler, agar sistem gerak dari lengan robot ini menjadi otomatis.

### 1.2 Permasalahan

Melihat latar belakang tersebut, maka dirumuskan permasalahan utama yaitu proses *etching* PCB secara manual yang tidak praktis,

menuntut untuk membuat alat yang dapat melakukan proses *etching* terhadap beberapa PCB secara otomatis.

### 1.3 Tujuan

Tujuan menuliskan Tugas Akhir ini adalah merancang dan membuat lengan robot 3 DOF dengan inverse kinematik untuk *etching* PCB.

### 1.4 Batasan Masalah

Dari permasalahan tersebut, batasan masalah pada proses *etching* PCB otomatis menggunakan lengan robot adalah :

1. Mesin ini bekerja berdasarkan ukuran PCB yang hendak dilarutkan sebagai acuan untuk menentukan lama waktu pelarutan.
2. Mikrokontroler yang digunakan adalah Arduino Nano dengan Motor servo yang digunakan dalam desain lengan robot penjepit pcb hanya motor servo jenis Standar 180°.
3. Lengan robot yang dibuat memiliki 3 link, 3 joint dan 1 *gripper*. *Gripper* digerakkan dengan satu motor untuk membuka dan menutup.
4. Lengan robot dibuat dengan material utama akrilik dan material lain seperlunya.

### 1.5 Metodologi Penelitian

Penelitian ini dilakukan melalui beberapa tahapan metodologi, yaitu, studi literatur, perancangan sistem, pengambilan data percobaan dan analisis data, dan yang terakhir adalah penyusunan laporan berupa buku Tugas Akhir.

Pada tahap studi literatur ini dilakukan pencarian data, spesifikasi dan bahan yang akan digunakan. setelah penentuan spesifikasi *hardware* yang akan digunakan, maka langkah selanjutnya adalah perancangan sistem yang akan dibuat. Pada tahap perancangan sistem terdiri dari dua yaitu, perancangan mekanik dan perancangan sistem elektrik. Pada tahap perancangan mekanik terdiri dari perancangan lengan robot. Sedangkan perancangan sistem elektrik terdiri dari Arduino nano, motor servo, *buck DC to DC converter*, dan *power supply*. Tahap selanjutnya adalah pengambilan data percobaan. Data percobaan yang telah diperoleh selanjutnya akan dianalisis. Dari hasil analisis, akan ditarik kesimpulan dari penelitian yang telah dilakukan. Tahap akhir penelitian adalah penyusunan laporan penelitian.



## **1.6 Sistematika Laporan**

Pembahasan Tugas Akhir ini akan dibagi menjadi lima Bab dengan sistematika sebagai berikut :

### **Bab I PENDAHULUAN**

Bab ini meliputi latar belakang, permasalahan, tujuan penelitian, metodologi penelitian, sistematika laporan, dan relevansi.

### **Bab II TEORI DASAR**

Bab ini menjelaskan tentang lengan robot, motor servo, Arduino nano, *buck DC to DC converter*, *switch-mode power supply*, *degrees of freedom*, serta kinematika robot.

### **Bab III PERANCANGAN SISTEM**

Bab ini membahas perencanaan dan pembuatan perangkat keras (*Hardware*) yang meliputi desain mekanik dan perangkat lunak (*Software*) yang meliputi program yang akan digunakan untuk menjalankan alat tersebut.

### **Bab IV HASIL SIMULASI DAN IMPLEMENTASI**

Bab ini memuat hasil simulasi dan implementasi serta analisis dari hasil tersebut.

### **Bab V PENUTUP**

Bab ini berisi kesimpulan dan saran dari hasil pembahasan yang telah diperoleh.

## **1.7 Relevansi**

Hasil yang diperoleh dari Tugas Akhir ini diharapkan memberikan manfaat berupa penerapannya pada dunia industri khususnya industri yang menggunakan PCB sebagai komponen produksi guna meningkatkan jumlah produksi dengan jumlah tenaga kerja yang sama dan diharapkan pengembangannya dalam dunia pendidikan untuk menutupi kekurangan agar alat dapat berfungsi optimal.

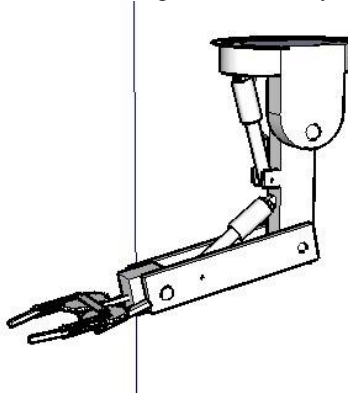
-----Halaman ini sengaja dikosongkan-----

## BAB II

### TEORI DASAR

#### 2.1 Lengan Robot

Robot lengan atau yang lebih dikenal dengan robot manipulator adalah salah satu jenis robot yang paling banyak digunakan di bidang industri. Sering dikatakan sebaliknya, sebuah robot manipulator industri umumnya sering disebut sebagai lengan robot dengan sistem mekanik yang terdiri dari rangka (*link*) dan sendi (*joint*). Robot manipulator diciptakan untuk berbagai keperluan dalam meningkatkan produksi sebagai pemindah barang dengan bentuk lengan-lengan kaku yang terhubung secara seri dan memiliki sendi yang dapat bergerak berputar (rotasi) atau memanjang/memendek (translasi atau prismatic), dengan kecepatan dan ketepatan yang akurat, dan pengendaliannya pun bisa berupa otomatis atau secara manual. Walaupun demikian, gerakan robot manipulator berbeda dari gerakan lengan manusia. Sendi robot manipulator memiliki kelebihan pada derajat kebebasan atau *Degree of Freedom* (DOF). Robot manipulator dapat bergerak bebas walaupun dari suatu sudut tertentu dengan kebebasan yang lebih dibanding manusia.



**Gambar 2.1** Lengan Robot 3 Dof

Lengan robot yang digunakan pada Gambar 2.1 memakai motor servo sebagai penggerak. Ada 5 motor servo yang digunakan untuk mengerjakan lengan robot. Ujung lengan robot berupa *gripper* yang digunakan untuk menjepit PCB.

### 2.1.1 Konsep Dasar Lengan Robot

Lengan robot biasanya berfungsi untuk mengambil suatu benda, kemudian meletakkan benda tersebut pada tempat lain yang bisa dilakukan secara manual maupun otomatis sesuai dengan kondisi yang diinginkan. lengan robot memiliki komponen-komponen sebagai pembentuknya seperti rangka dan sendi, juga memiliki beberapa istilah dalam pengukurannya. Beberapa istilah dan definisi yang banyak digunakan pada lengan robot adalah :

1. *Link*  
*Link* merupakan salah satu bagian dari kerangka robot yang dihubungkan secara bersamaan untuk membentuk sebuah rangkaian kinematik.
2. *Joint* (Sendi)  
*Joint* adalah penghubung antar *link* yang dapat menentukan pergerakan *relative* yang terbatas.
3. *End-effector*  
Merupakan suatu komponen pada ujung lengan robot yang mempunyai fungsi menghubungkan lengan robot dengan objek yang akan dijadikan kerja dari robot. Sebagai contoh *End-Effector* diantara lain *Gripper* dan *Tools*. Namun pada Laporan Akhir ini menggunakan *Gripper* sebagai *End Effector Robot*.
4. *Work Space*  
Adalah luas total ruang kerja yang dapat dijangkau oleh *End-Effector* ketika lengan robot melakukan semua gerakan yang mungkin.
5. Akurasi  
Pengukuran atas seberapa dekat sebuah manipulator dapat mencapai titik tujuan yang diinginkan pada ruang kerjanya.
6. *Repeatability*  
Pengukuran atas seberapa dekat sebuah manipulator dapat kembali mencapai titik tujuan sebelumnya.
7. Derajat kebebasan  
jumlah gerakan yang dapat dibuat oleh suatu objek terhadap sistem koordinat yang dapat menyebabkan perubahan posisi.
8. Poros Gerakan  
Adalah mekanisme yang memungkinkan robot untuk bergerak secara lurus atau berotasi.

### 2.1.2 Bagian-Bagian Lengan Robot

1. *Mechanical Arm*

Adalah bagian dasar dari konstruksi lengan robot yang dikendalikan pergerakannya agar dapat membentuk lengan robot sesuai posisi yang diinginkan.

2. *End-Effector*

Kemampuan robot juga tergantung pada piranti yang dipasang pada lengan robot. Piranti ini dikenal dengan *end effector*. *End effector* ada dua jenis yaitu Pencengkram (*gripper*) yang digunakan untuk memegang dan menahan objek dan peralatan (*tool*) yang digunakan untuk melakukan operasi tertentu pada suatu objek. Contohnya: bor, penyemprot cat, gerinda, las dan sebagainya.

3. Penggerak (*Actuator*)

Bagian lengan robot yang digunakan untuk mekanisme penggerak lengan robot. Pada lengan robot penjepit PCB ini menggunakan penggerak berupa motor servo. Aktuator juga dapat berupa hidrolik ataupun pneumatik yang digunakan untuk mengendalikan persendian karena dapat menghasilkan gerakan linier secara langsung atau pula aktuator motor listrik yang menghasilkan gerakan rotasi. Penggerak yang umum digunakan pada saat ini adalah penggerak motor servo karena lebih mudah dikontrol dibanding penggerak lainnya.

4. Pengendali (*Controller*)

Pengendali adalah mekanisme (baik secara perangkat keras maupun perangkat lunak) yang digunakan untuk mengatur seluruh pergerakan atau proses yang dilakukan lengan robot.

### 2.2 Motor Servo

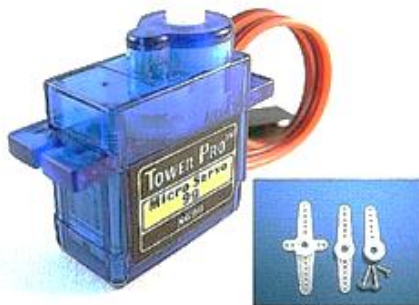
Motor servo merupakan sebuah motor DC kecil yang diberi sistem *gear* dan potensiometer sehingga dapat menempatkan horn servo pada posisi yang diinginkan. Motor servo prinsipnya mempunyai sistem *close loop*, di mana posisi dari motor akan di informasikan kembali ke rangkaian *control* yang ada di dalam motor servo.

Berbeda dengan motor DC, motor servo tidak bergerak kontinu, melainkan menuju sudut tertentu saja dan berhenti di sudut tersebut. Motor ini digunakan untuk aplikasi gerakan-gerakan sudut dari robot, contohnya gerakan lengan, *gripper* menjepit benda, atau gerakan kaki melangkah.<sup>[2]</sup>

Di lihat dari putarannya, terdapat dua jenis motor servo yaitu motor servo *standard* dan motor servo *continuous*. Motor servo *continuous* dapat bergerak sampai satu putaran penuh sebesar 360 derajat, sedangkan motor servo *standard* hanya dapat bergerak sampai setengah putaran yaitu 180 derajat. Pada pembuatan lengan robot penjepit PCB ini menggunakan motor servo standar 180°. Motor servo jenis ini merupakan motor yang hanya mampu bergerak dua arah dan mempunyai defleksi masing-masing sudut mencapai 90° sehingga total defleksi sudut dari kanan, tengah dan kiri adalah 180°. Lengan robot penjepit PCB ini menggunakan 5 buah motor servo yang memiliki perbedaan spesifikasinya. Servo 1, 2 dan 3 menggunakan motor servo jenis MG995. Servo 4 dan 5 menggunakan servo ukuran kecil jenis SG90. Bentuk fisik kedua jenis motor servo tersebut dapat dilihat pada Gambar 2.2 dan Gambar 2.3 serta spesifikasinya yang terdapat pada Tabel 2.1 dan Tabel 2.2



**Gambar 2.2** Bentuk Fisik Motor Servo MG995<sup>[3]</sup>



**Gambar 2.3** Bentuk Fisik Motor Servo SG90<sup>[4]</sup>

**Tabel 2.1** Spesifikasi Motor Servo MG995

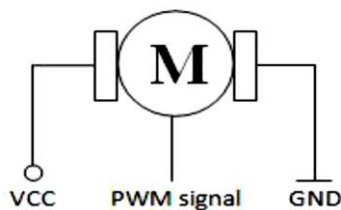
<i>Operating Voltage</i>	4.8 V ~ 7.2 V
<i>Operating Current</i>	100 mA
<i>Operating Speed</i>	0.17 sec/60° (4.8V) ~ 0.13 sec/60° (6.0V)
<i>Torque</i>	13.0 kg/cm (4.8V) ~ 15.0kg/cm (6.0V)
<i>Dead Band Width</i>	4 $\mu$ s
<i>Temperature Range</i>	-30~+60°C

**Tabel 2.2** Spesifikasi Motor Servo SG90

<i>Operating Voltage</i>	4.8 V ~ 6 V
<i>Operating Current</i>	100 mA
<i>Operating Speed</i>	0.12 sec/ 60° (4.8V)
<i>Torque</i>	1.2 kg/cm ~ 1.4 kg/cm (4.8V)
<i>Dead Band Width</i>	7 $\mu$ s
<i>Temperature Range</i>	-30~+60°C

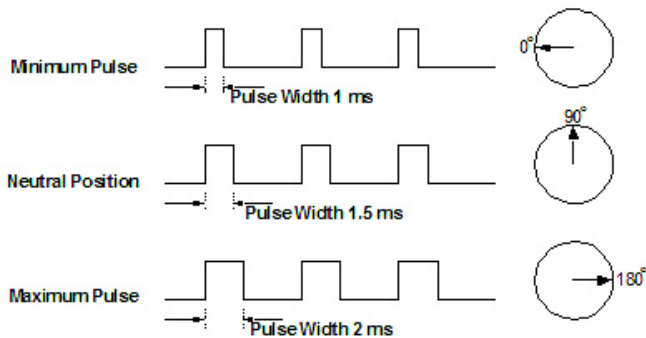
### 2.2.1 Konstruksi Motor Servo

Motor Servo memiliki 3 kabel yaitu *orange* sebagai I/O pin, merah sebagai Vcc dan coklat sebagai *ground*. Dengan demikian kita dapat mengontrol motor servo melalui kabel I/O yang berwarna *orange*. Pada Gambar 2.4 dibawah ini merupakan pin-pin dan pengkabelan dari motor servo yang dihubungkan pada rangkaian pengontrol.



**Gambar 2.4** Pin Out Kabel Motor Servo

Operasional motor servo dikendalikan oleh sebuah sinyal berperiode  $\pm 20$  ms, dimana lebar pulsa antara 1 ms dan 2 ms menyatakan akhir dari jangkauan sudut maksimum. Jika motor servo diberikan pulsa dengan lebar 1,5 ms maka posisi yang dihasilkan adalah  $90^\circ$ . Jika diberikan pulsa kurang dari 1,5 ms maka posisi mendekati  $0^\circ$ . Jika diberikan pulsa lebih dari 1,5 ms maka posisi mendekati  $180^\circ$ .<sup>[5]</sup>



**Gambar 2.5** Pulsa Kendali Motor Servo<sup>[6]</sup>

**Tabel 2.3** Nilai Penyisalan Motor Servo

Input Pulsa	Derajat
1 ms	0
1,25 ms	45
1,5 ms	90
1,75 ms	135
2 ms	180

Pada dasarnya penggunaan servo menggunakan cara yang sama, yaitu dengan memberikan lebar pulsa tertentu pada sudut putarnya. Untuk servo standard, sudut putarnya adalah 180 derajat yang dapat dioperasikan dalam dua arah (*clock wise / counter clock wise*). Pada Gambar 2.5 diatas merupakan lebar pulsa yang dibutuhkan untuk mengoperasikan motor servo *standard*. Pulsa diatas harus diberikan secara terus menerus, agar motor servo mempertahankan posisinya sesuai dengan pulsa yang diberikan. Untuk nilai penyisalan itu sendiri dapat dilihat pada Tabel 2.3.

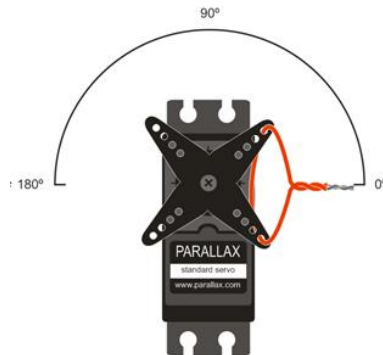


### 2.2.2 Prinsip Kerja Motor Servo

Seperti yang sudah kita ketahui pada karakteristik motor servo, pemberian besar pulsa dari mikrokontroler menentukan besar sudut yang harus dilakukan oleh motor servo. Pemberian pulsa dari mikrokontroler ke motor servo tersebut tidak memerlukan rangkaian driver tambahan, karena di dalam sebuah motor servo sudah terdapat *internal gear* dan rangkaian driver yang memungkinkan motor servo dapat langsung dihubungkan ke mikrokontroler. Pengaturan sudut motor servo diperlukan untuk mengetahui gerakan dari motor servo dan pulsa yang harus diberikan ke motor servo dalam pergerakannya ke kanan atau ke kiri. Dari pulsa yang kita berikan kita dapat melihat gerakan dari motor servo.

Motor servo dikendalikan dengan cara mengirimkan sebuah pulsa yang lebar pulsanya bervariasi. Pulsa tersebut dimasukkan melalui kabel kontrol motor servo. Sudut atau posisi motor servo akan diperoleh dari lebar pulsa. Biasanya lebar pulsanya antara  $20\mu\text{s}$  sampai  $100\mu\text{s}$  dengan periode pulsa sebesar  $20\text{ms}$ . Lebar pulsa akan mengakibatkan perubahan posisi pada servo. Misalnya sebuah pulsa  $50\mu\text{s}$  akan memutar motor pada posisi  $90^\circ$  (posisi netral). Agar posisi servo tetap pada posisi ini, maka pulsa harus terus diberikan pada servo.

Ketika sebuah pulsa yang dikirim ke servo kurang dari  $50\mu\text{s}$ , servo akan berputar *counter clock wise* menuju ke posisi tertentu dari posisi netral. Jika pulsa yang dikirim lebih dari  $50\mu\text{s}$ , servo akan berputar *clock wise* menuju ke posisi tertentu dari posisi netral. Arah putaran motor servo dapat dilihat pada Gambar 2.6. Setiap servo memiliki spesifikasi lebar pulsa minimum dan maksimum sendiri-sendiri, tergantung jenis dan merk servo. Umumnya antara  $20\mu\text{s}$  sampai  $100\mu\text{s}$ .



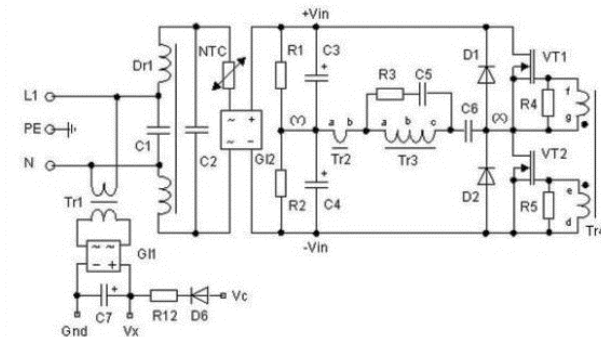
**Gambar 2.6** Arah Putaran Motor Servo Standard<sup>[7]</sup>

### 2.3 Switch-Mode Power Supply

*Switch-Mode Power Supply* (SMPS) adalah jenis *Power Supply* yang langsung menyearahkan (*rectify*) dan menyaring (*filter*) tegangan Input AC untuk mendapatkan tegangan DC. Tegangan DC tersebut kemudian di-switch *ON* dan *OFF* pada frekuensi tinggi dengan sirkuit frekuensi tinggi sehingga menghasilkan arus AC yang dapat melewati Transformator Frekuensi Tinggi.

*Power Supply* dengan regulator *switching* ini lebih dikenal sebagai *power supply switching*. Kelebihan *power supply switching* adalah efisiensi daya yang besar dibandingkan dengan *power supply* dengan regulasi biasa. Selain itu berat lebih ringan dan ukuran lebih kecil karena menggunakan frekwensi diatas 20Khz. Semakin tinggi frekwensi switching, maka ukuran trafo dan kapasitor filter semakin kecil. Gambar rangkaian *power supply switching* dapat dilihat pada Gambar 2.7

Prinsip kerja *power supply switching* ini yaitu tegangan masukan dari listrik AC 220V disearahkan menjadi tegangan DC menggunakan *diode bridge* dan 3 buah *elco filter* besar. Tegangan masukan DC dicacah dengan menggunakan “*power switch on-off*” sehingga menghasilkan tegangan pulsa-pulsa DC dengan frekwensi tinggi. Tegangan DC yang telah dicacah mempunyai karakteristik seperti tegangan AC sehingga dapat dilewatkan sebuah trafo atau induktor untuk dinaikkan ataupun diturunkan tegangannya. Tegangan keluaran dari trafo masih berupa pulsa-pulsa frekwensi tinggi dan kemudian dirubah menjadi tegangan dc menggunakan dioda penyearah dan *filter elco* lalu diberi *loop* umpan balik agar tegangan keluaran stabil.



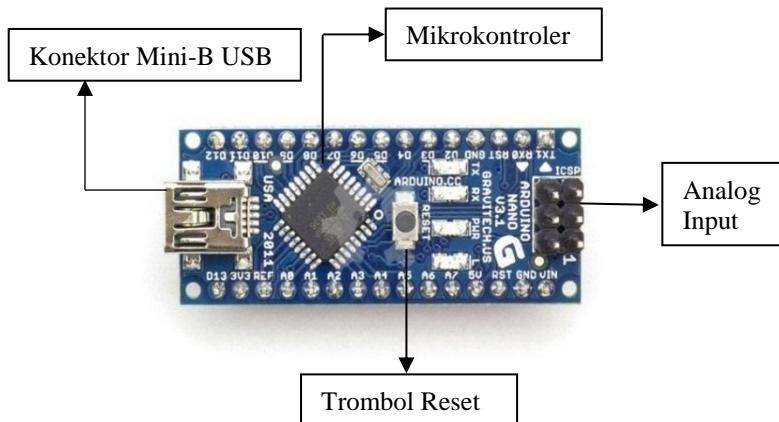
**Gambar 2.7** Gambar Rangkaian Switch Mode Power Supply<sup>[8]</sup>

## 2.4 Arduino Nano

Arduino nano adalah papan mikrokontroler berbasis ATmega328 dirancang untuk memudahkan penggunaan elektronik dalam berbagai bidang. Arduino nano mempunyai 14 *pin* masukan dan keluaran digital (termasuk 6 diantaranya dapat berfungsi sebagai keluaran *Pulse Width Modulation*), 6 masukan *analog*, sebuah kristal 16 MHz, sebuah tombol *reset* dan konektor Mini-B USB. Arduino nano beroperasi pada tegangan 5 volt.

### 2.4.1 Perangkat Keras Arduino

Dalam Tugas Akhir alat lengan robot penjepit PCB 3 dof berbasis Arduino yang ditunjukkan pada Gambar 2.8 ini menggunakan jenis Arduino yang digunakan adalah Arduino Nano. Arduino Nano adalah board Arduino terkecil, menggunakan mikrokontroler Atmega 328 untuk Arduino Nano 3.x dan Atmega168 untuk Arduino Nano 2.x. Varian ini mempunyai rangkaian yang sama dengan jenis Arduino Duemilanove, tetapi dengan ukuran dan desain PCB yang berbeda. Arduino Nano tidak dilengkapi dengan soket catudaya, tetapi terdapat pin untuk catu daya luar atau dapat menggunakan catu daya dari mini USB port. Arduino cukup mudah digunakan karena pemogramannya *open source*. Fungsi konfigurasi Arduino terhadap lengan robot ditunjukkan pada Tabel 2.4.



**Gambar 2.8** Bagian – Bagian Arduino Nano

**Tabel 2.4** Fungsi Konfigurasi Arduino Nano

Nama Pin	Fungsi	Keterangan
D10	Tx (Pengirim)	Komunikasi Serial ke Arduino <i>Slider</i>
D11	Rx (Penerima)	Komunikasi Serial ke Arduino <i>Slider</i>
D2	Servo Dof 1	Mengirimkan Data ke Servo Dof 1
D3	Servo Dof 2	Mengirimkan Data ke Servo Dof 2
D4	Servo Dof 3	Mengirimkan Data ke Servo Dof 3
D5	Servo Pemutar	Mengirimkan Data ke Servo Pemutar
D6	Servo Gripper	Mengirimkan Data ke Servo <i>Gripper</i>
VCC	<i>Supply</i> Tegangan	+5V Input dari <i>Power Supply</i>
GND	<i>Supply</i> Ground	Input Ground dari <i>Power Supply</i>

### 2.4.2 Spesifikasi Arduino Nano

Arduino nano memiliki spesifikasi dalam perangkatnya. Spesifikasi Arduino nano dijelaskan dalam Tabel 2.5

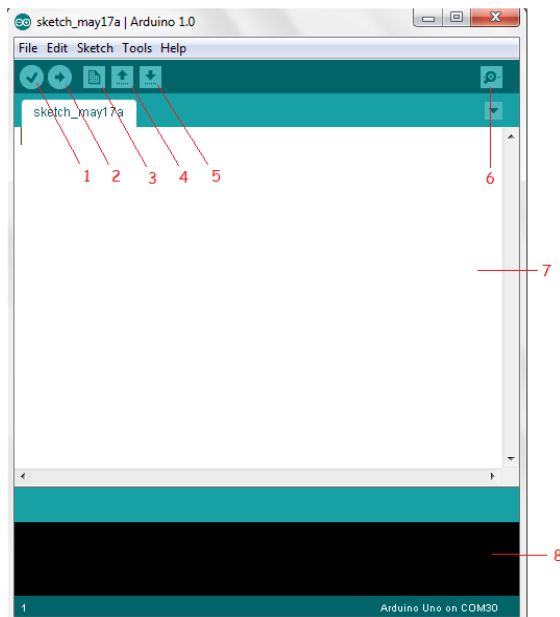
**Tabel 2.5** Spesifikasi Arduino Nano

Mikrokontroler	Atmel ATmega 168 / ATmega 328
Tegangan Operasi	5 Volt
<i>Input Voltage</i> (disarankan)	7-12 Volt
<i>Input Voltage</i> (limit)	6-20 Volt
Pin Digital I/O	14 (6 pin sebagai output PWM)
Pin Input Analog	8
Arus DC per pin I/O	40 mA
<i>Flash Memory</i>	16 kB (ATmega 168 ) / 32 kB (ATmega 328)
SRAM	1 kB (ATmega 168 ) / 2 kB (ATmega 328)
EEPROM	512 byte (ATmega 168 ) / 1 kB (ATmega 328)
<i>Clock Speed</i>	16 MHz
Dimensi	45 mm x 18 mm
Berat	5 g

### 2.4.3 IDE (*Integrated Deveolpment Environment*)







Pemrograman board Arduino Nano dilakukan dengan menggunakan Arduino Software (IDE). *Chip* ATmega328 yang terdapat pada Arduino Nano telah diisi program awal yang sering disebut *bootloader*. *Bootloader* tersebut yang bertugas untuk memudahkan melakukan pemrograman lebih sederhana menggunakan Arduino Software tanpa harus menggunakan tambahan *hardware* lain.

IDE *Integrated Development Environment* merupakan *software* pemrograman yang beroperasi di komputer untuk menghasilkan sebuah file berformat hex yang akan di-download pada papan arduino atau papan sistem mikrokontroler lainnya. Pemrograman ini tergolong mudah dan mampu membuat pengguna lebih cepat dalam menguasai dan mempelajarinya. Tampilan awal pada IDE terdapat pada Gambar 2.9 Tampilan ini berupa *workspace* awal dari Arduino IDE. Dan akan dijelaskan juga fungsi-fungsi yang ada pada tampilan awal IDE ini. Seperti yang ada pada tampilan awal tersebut terdapat *verify*, *upload*, *new*, *open*, *save*, *serial monitor*, *sketch page*, *status page*.



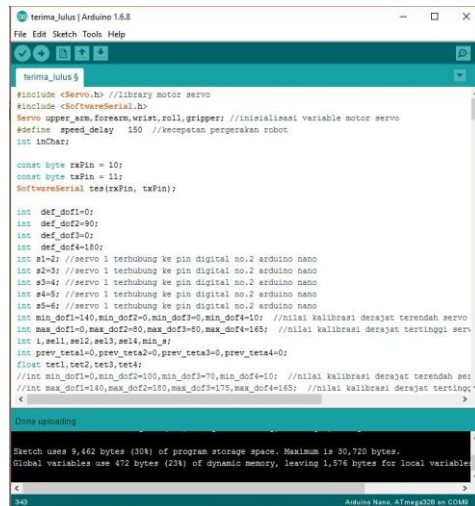
**Gambar 2.9** Tampilan Awal IDE Arduino

Keterangan mengenai tampilan IDE pada Gambar 2.9 adalah sebagai berikut :

1.  Adalah Ikon Menu **Verify** yang bergambar ceklis. *Verify*, berfungsi untuk melakukan pengujian pada program atau *sketch* yang telah dibuat. Apabila *sketch* sudah benar, maka *sketch* tersebut akan dikompilasi. Kompilasi adalah proses mengubah kode program kedalam kode mesin.
2.  Adalah Ikon Menu **Upload** yang bergambar panah ke arah kanan. *Upload*, berfungsi mengirimkan kode mesin hasil kompilasi ke *board* Arduino.
3.  Merupakan ikon *create new project*, berfungsi membuka *sketch* baru.
4.  Adalah Ikon Menu **Open** yang bergambar panah ke arah atas. *Open*, berfungsi membka *sketch* yang pernah dibuat dan membuka kembali untuk di-*edit* ataupun di-*upload* ulang.
5.  Adalah ikon **Save** yang bergambar panah ke arah bawah. *Save*, berfungsi menyimpan program yang ditulis di *sketch*.
6.  Adalah Ikon Menu **Serial Monitor** yang bergambar kaca pembesar (*loop*). *Serial Monitor*, berfungsi menampilkan data yang dikirim dan diterima melalui komunikasi serial.
7. **Sketch Page**, berfungsi sebagai tempat untuk menulis program.
8. **Status Page**, berfungsi mengetahui status proses ketika program telah dikompilasi atau di-*upload*

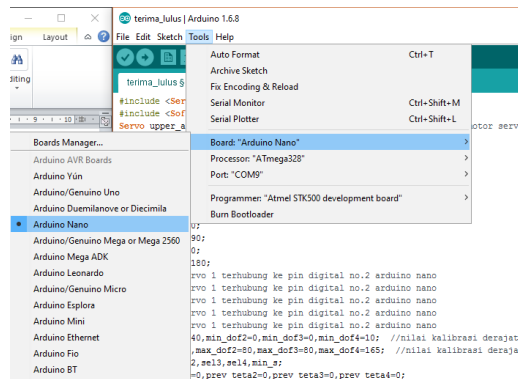
Setelah mengetahui beberapa ikon yang digunakan dan telah membuat program dalam *software*, maka langkah selanjutnya yang harus dilakukan adalah cara mengunggah program pada Arduino. Untuk mengunggah program pada Arduino terdapat langkah-langkah yang harus dilakukan. Berikut langkah-langkah mengunggah program pada Arduino:

1. Langkah pertama yang dilakukan setelah membuat program adalah melakukan *verify* program yang digunakan dengan cara menekan tombol *verify*. Jika tidak ada *error* maka akan muncul kata *done compiling* seperti Gambar 2.10 di bawah ini.



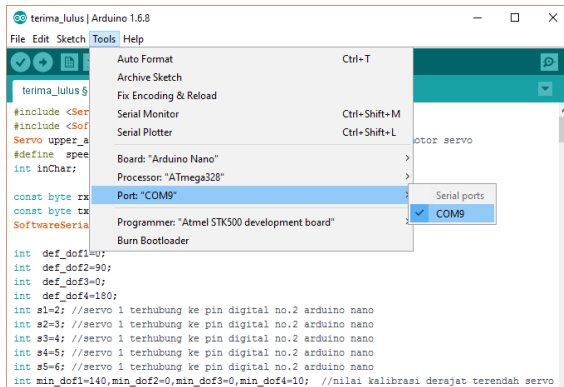
**Gambar 2.10** Tampilan Arduino Ketika Tidak Terjadi *Error* Pada Saat *Verify Program*

2. Langkah kedua adalah menghubungkan *hardware* Arduino dengan laptop atau pc yang digunakan.
3. Langkah ketiga adalah memilih *board* Arduino yang digunakan dengan cara menekan *Tools* lalu *board* dan *board* yang digunakan seperti pada Gambar 2.11 dibawah ini.



**Gambar 2.11** Memilih *Board* Arduino Pada Arduino

4. Langkah keempat adalah memilih *port* yang digunakan pada laptop atau PC seperti pada Gambar 2.12 di bawah ini.



**Gambar 2.12** Memilih *Port* Pada Arduino

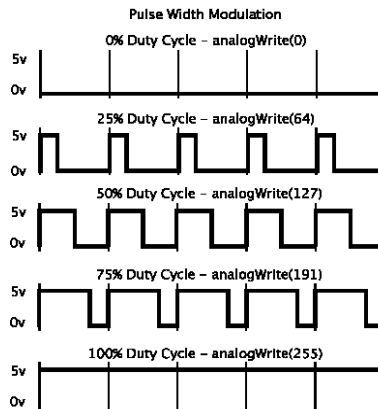
5. Langkah terakhir adalah menekan *ikon upload* atau menekan *file* → *upload*.

#### 2.4.4 Sinyal PWM pada Arduino

PWM (*Pulse Width Modulation*) pada Arduino adalah metode untuk mendapatkan bentuk sinyal *analog* dari sinyal *digital* dengan frekuensi tertentu. PWM pada Arduino bekerja pada frekuensi 500 Hz, artinya terdapat 500 ketukan dalam satu detik. PWM pada Arduino dituliskan dalam bentuk desimal dengan jangkauan angka biner 8 bit. Jadi, jika ditulis dalam desimal, maka bernilai 0 untuk minimum atau 255 pada keluaran maksimum. Ketika suatu pin keluaran PWM diberi nilai 0, maka pin tersebut akan memberikan keluaran 0 volt sebanyak 0 ketukan dalam 1 detik atau setara dengan GND. Jika suatu pin PWM diberi nilai 255, maka pin tersebut akan memberikan keluaran 5 volt sebanyak 500 ketukan dalam 1 detik. Salah satu contoh penggunaan sinyal PWM adalah untuk memberikan sinyal aksi kepada motor servo. Gambar 2.13 mengilustrasikan sinyal keluaran yang dibangkitkan oleh suatu pin PWM.

Pada Arduino IDE, ATmega328 dapat membangkitkan sinyal PWM dengan menuliskan *analogWrite* (bilangan PWM) dengan terlebih dahulu mendefinisikan pin yang dimaksud. Pin yang bisa dimanfaatkan untuk PWM pada Arduino nano diantaranya adalah Pin D4, pin D5, pin D6, pin D9, pin D10 dan pin D11.





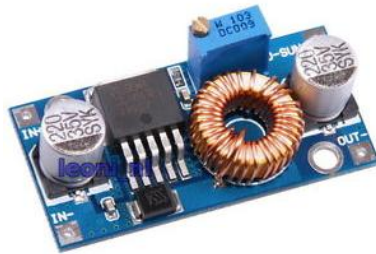
**Gambar 2.13** Sinyal PWM dengan Nilai yang Diberikan<sup>[5]</sup>

## 2.5 Buck DC to DC Converter

Konverter DC-DC berfungsi untuk mengkonversikan daya listrik searah (DC) ke bentuk daya listrik DC lainnya yang terkontrol arus, atau tegangan, atau dapat juga terkontrol arus dan tegangan. Konverter jenis *buck* merupakan konverter penurun tegangan yang mengkonversikan tegangan masukan DC menjadi tegangan DC lainnya yang lebih rendah. Seperti rangkaian konverter *DC to DC* tipe *buck* yang dapat dilihat pada Gambar 2.14, rangkaian ini terdiri atas satu saklar aktif (MOSFET), satu saklar pasif (diode), kapasitor dan induktor sebagai tapis keluarannya.

Buck-converter memanfaatkan sifat induktor terhadap guncangan listrik berfrekwensi tinggi dan bekerja dengan adanya denyut-denyut tegangan (sebagaimana layaknya SMPS). Karena itu di dalam sebuah rangkaian buck-converter selalu terdapat generator sinyal, transistor penguat, dioda, kondensator dan induktor.

Prinsip kerja *Buck-Converter* adalah dengan menggunakan *switch* yang bekerja secara terus-menerus (*ON-OFF*). *switch* pada *Buck-Converter* akan bekerja secara terus-menerus. Kecepatan *switch* (dalam realisasinya) akan tergantung pada *Duty Cycle* dan frekuensi yang digunakan.

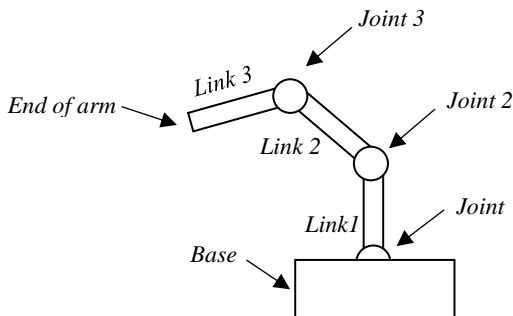


**Gambar 2.14** Rangkaian Konverter DC-DC Tipe Buck<sup>[9]</sup>

## 2.6 Degrees of Freedom

Teknologi lengan robot dapat digunakan sebagai alat bantu manusia yang dapat digunakan pada tempat-tempat yang tidak memungkinkan untuk di jangkau atau berbahaya bagi manusia. Dalam lengan robot dikenal dengan istilah *Degree of Freedom* (DOF) atau derajat kebebasan yang merupakan setiap titik sumbu gerakan mekanik pada lengan robot.

Lengan robot dapat digambarkan dengan derajat kebebasannya. Jumlah DOF biasanya mengacu pada jumlah sendi yang dimiliki lengan robot. Dimana semakin banyak DOF mengindikasikan peningkatan fleksibilitas dalam memposisikan alat. Setiap sendi dihubungkan dengan dua batang hubung, yaitu batang hubung masukan dan keluaran. Lengan robot penjepit PCB ini dirancang untuk memposisikan *end-effector* dengan 3 derajat kebebasan seperti pada Gambar 2.15.



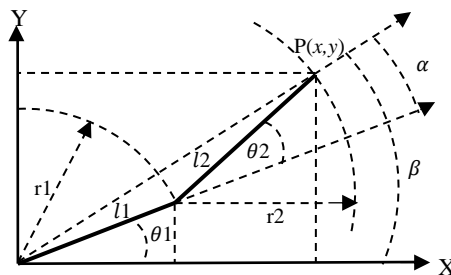
**Gambar 2.15** Ilustrasi Lengan Robot 3 Sendi

## 2.7 Kinematika Robot <sup>[10]</sup>

Untuk pergerakan robot diterapkan kinematika robot. Kinematika robot ini dapat didefinisikan sebagai pergerakan robot terhadap sistem kerangka koordinat acuan yang diam atau bergerak tanpa memperhatikan gaya ataupun faktor lain yang mempengaruhi gerakan robot tersebut. Model kinematika merepresentasikan hubungan *end-effector* dalam ruang tiga dimensi dengan variabel sendi dalam ruang sendi. Kinematika pada robot secara umum terbagi menjadi dua yakni *Forward Kinematic* dan *Inverse Kinematic*. *Forward Kinematic* adalah metode penentuan dari posisi akhir ujung lengan robot dengan diketahui sudut masing-masing *joint* dan dicari koordinat cartesius dari ujung lengan robot. Dapat dimisalkan sebuah robot n-DOF diketahui sudut dari tiap *joint* maka dapat digunakan metode *Forward Kinematic* untuk mendapatkan kordinat posisi robot (kordinat posisi x,y,z). Sedangkan *Inverse kinematic* adalah metode penentuan posisi akhir dari ujung lengan robot dengan diketahui koordinat x, y, dan z sedangkan yang dicari adalah sudut dari masing-masing *joint*. Dengan membalik *forward kinematik*, dapat didefinisikan persamaan dari kinematika terbalik atau *inverse kinematic*.

Pada pembuatan lengan robot penjepit pcb ini menggunakan algoritma *inverse kinematic* sehingga tidak perlu lagi mengatur kombinasi besar sudut-sudut engsel pada robot, namun cukup memanipulasi posisi koordinat ujung lengan atau bagian lain dari lengan robot untuk mencapai suatu gerak tertentu dari lengan robot. Hal ini dikarenakan perangkat lunak lengan robot yang telah diberi algoritma ini akan secara otomatis menghitung kombinasi nilai sudut-sudut engsel untuk setiap input berupa informasi letak koordinat titik yang diberikan.

### 2.7.1 Kinematik Robot Lengan Dua Sendi



**Gambar 2.16** Konfigurasi Robot 2 Sendi (2 DOF)

Langkah pertama untuk mencari nilai kinematik robot 2 sendi yaitu dengan memodelkan pergerakan seperti pada Gambar 2.16. Kedudukan ujung lengan dinyatakan sebagai  $P(x,y)$ .

$$P(x,y) = f(\theta_1, \theta_2) \quad (1)$$

Jika  $P$  diasumsikan sebagai vektor penjumlahan yang terdiri dari vector  $r$  lengan-1 dan  $r_2$  lengan-2,

$$r_1 = [l_1 \cos \theta_1, l_1 \sin \theta_1] \quad (2)$$

$$r_2 = [l_2 \cos (\theta_1 + \theta_2), l_2 \sin (\theta_1 + \theta_2)] \quad (3)$$

maka

$$x = l_1 \cos \theta_1 + l_2 \cos (\theta_1 + \theta_2) \quad (4)$$

$$y = l_1 \sin \theta_1 + l_2 \sin (\theta_1 + \theta_2) \quad (5)$$

Persamaan (4) dan (5) adalah persamaan *forward kinematic* dari lengan robot 2 sendi.

*Inverse Kinematic* robot dapat dijabarkan sebagai berikut. Dengan menggunakan hukum identitas trigonometri,

$$\cos (a + b) = \cos(a) \cos(b) - \sin(a) \sin(b) \quad \text{dan} \quad (6)$$

$$\sin (a + b) = \sin(a) \cos(b) + \sin(b) \cos(a) \quad \text{dan} \quad (7)$$

Persamaan (4) dan (5) dapat ditulis kembali,

$$x = l_1 \cos \theta_1 + l_2 \cos \theta_1 \cos \theta_2 - l_2 \sin \theta_1 \sin \theta_2 \quad (8)$$

$$y = l_1 \sin \theta_1 + l_2 \sin \theta_1 \cos \theta_2 + l_2 \cos \theta_1 \sin \theta_2 \quad (9)$$

Dari dua persamaan terakhir ini kita dapat mencari  $\theta_2$  terlebih dahulu dengan menggunakan  $\cos \theta_2$  dari kedua persamaan. Dengan operasi pangkat dua pada kedua nya, dan dikombinasikan didapat,

$$\cos \theta_2 = \frac{x^2 + y^2 - l_1^2 - l_2^2}{2l_1 l_2} \quad (10)$$

sehingga

$$\theta_2 = \arccos\left(\frac{x^2 + y^2 - l_1^2 - l_2^2}{2l_1l_2}\right) \quad (11)$$

Perhatikan kembali Gambar 2.16. Sudut  $\theta_1$  dapat dicari melalui,

$$\tan \alpha = \frac{l_2 \sin \theta_2}{l_2 \cos \theta_2 + l_1} \text{ dan } \tan \beta = \frac{y}{x} \quad (12)$$

Sedangkan

$$\theta_1 = \beta - \alpha \quad (13)$$

Dengan menggunakan hukum identitas trigonometri,

$$\tan(a - b) = \frac{\tan(a) - \tan(b)}{1 + \tan(a) \tan(b)} \quad (14)$$

didapat

$$\tan \theta_1 = \frac{y(l_1 + l_2 \cos \theta_2) - x \cdot l_2 \sin \theta_2}{x(l_1 + l_2 \cos \theta_2) + y \cdot l_2 \sin \theta_2} \quad (15)$$

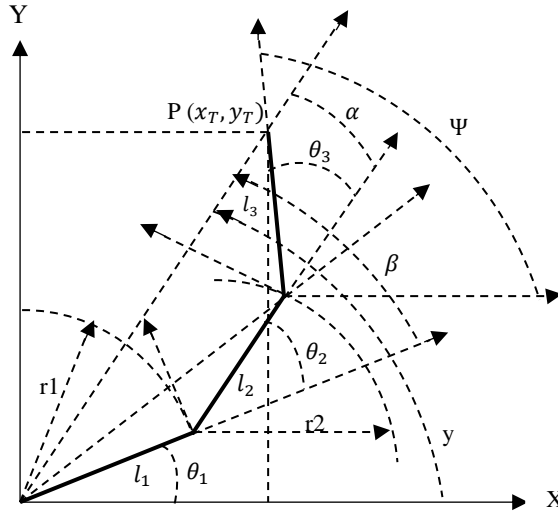
sehingga  $\theta_1$  dapat dihitung,

$$\theta_1 = \arctan \left( \frac{y(l_1 + l_2 \cos \theta_2) - x \cdot l_2 \sin \theta_2}{x(l_1 + l_2 \cos \theta_2) + y \cdot l_2 \sin \theta_2} \right) \quad (16)$$

Walhasil, persamaan (11) dan (16) adalah persamaan *inverse kinematic* bagi lengan robot dua sendi.

### 2.7.2 Kinematik Robot Lengan Tiga Sendi

Sekarang kita akan membahas kinematik dari robot lengan tiga sendi. Daerah kerja robot ini adalah 2D seperti pada robot lengan dua sendi yang telah dijelaskan sebelumnya. Konfigurasinya ditunjukkan pada Gambar 2.17 berikut ini.



**Gambar 2.17** Konfigurasi Robot Tiga Sendi (3 DOF)

Dengan cara analisis *forward kinematic* yang sama seperti pada persamaan (3) hingga (8) koordinat  $P(x_T, y_T)$  dapat diperoleh,

$$X_t = l_1 \cos \theta_1 + l_2 \cos (\theta_1 + \theta_2) + l_3 \cos (\theta_1 + \theta_2 + \theta_3) \quad (17)$$

$$Y_t = l_1 \sin \theta_1 + l_2 \sin (\theta_1 + \theta_2) + l_3 \sin (\theta_1 + \theta_2 + \theta_3) \quad (18)$$

Dengan

$$\psi = (\theta_1 + \theta_2 + \theta_3) \quad (19)$$

$\psi$  adalah sudut arah terhadap lengan-3 terhadap sumbu X.

Perhatikan bahwa koordinat P dapat dicapai dalam lebih dari satu konfigurasi  $\psi = (\theta_1 + \theta_2 + \theta_3)$ . Sebagai misal, katakanlah ujung lengan pada posisi P kita tahan dengan tangan pada satu kedudukan yang tetap, kemudian sendi-2 dan sendi-3 kita goyang, maka konfigurasi sudut di sendi 1, 2 dan 3 akan dapat bergerak dengan ujung koordinat P tetap pada kedudukannya. Dari sinilah dikatakan, jika arah  $\psi$  tidak diperhitungkan maka robot ini memiliki fungsi kinematik yang redundant

(berlebih) Karena penyelesaian persamaan untuk mendapatkan  $(\theta_1 + \theta_2 + \theta_3)$  dari suatu P adalah tidak tunggal (lebih dari satu penyelesaian). Sifat redundant (*redundancy*) ini dapat mengurangi derajat kebebasan robot. Robot 3DOF yang redundant berfungsi sama seperti 2DOF jika hanya berorientasi pada koordinat P saja.

Koordinat P  $(x_T, y_T)$  juga dapat dihitung dengan memanfaatkan hasil dari perhitungan pada *forward* kinematic robot lengan 2 sendi pada persamaan (8) dan (9) dengan rumus sebagai berikut.

$$x_T = x - l_3 \cos \psi \quad (20)$$

$$y_T = y - l_3 \sin \psi \quad (21)$$

Untuk *inverse kinematic*, jika  $(x_T, y_T)$  dan  $(x, y)$  diketahui maka  $\theta_2$  dan  $\theta_1$  dapat dicari dengan menggunakan persamaan (20) dan (21). Dari  $(x_T, y_T)$  dan  $(x, y)$ ,  $\psi$  juga dapat dicari, sehingga  $\theta_3$  dapat ditentukan.

## 2.8 Larutan H3 Pengganti *FeCl3* untuk *Etching* PCB

Proses *etching* merupakan pelarutan tembaga yang tidak dikehendaki dalam pembuatan jalur pada PCB. Umumnya dalam proses *etching*, larutan *FeCl3* atau juga dapat disebut *Ferri Chloride* lebih sering digunakan karena mudah di dapatkan dan harga relatif murah. Akan tetapi *FeCl3* mempunyai banyak kekurangan diantaranya proses pelarutan tembaga yang lama serta meninggalkan bekas warna yang pekat dan susah dihilangkan.

Dalam proses *etching* PCB dengan lengan robot penjepit PCB ini menggunakan cairan pelarut pengganti yaitu *Hydrochloride Acid* (*HCL*), *Hydrogen Peroxide* ( $H_2O_2$ ) dan Air ( $H_2O$ ). Hal ini dikarenakan pelarutan tembaga lebih cepat dibanding menggunakan pelarut *FeCl3*. Gambar larutan yang dipakai dapat dilihat pada Lampiran B. Reaksi kimia yang terjadi saat pelarutan tembaga dengan bahan pelarut ini adalah  $Cu + 2HCl + H_2O_2 \longrightarrow CuCl_2 + 2H_2O$  Artinya, oksigen ekstra dalam larutan dari peroksida mengoksidasi logam tembaga, dengan adanya asam, untuk membuat tembaga (II) klorida. *CuCl2* yang dihasilkan menjadi warna hijau zamrud.

Setelah melarutkan banyak tembaga ke dalam cairan pelarut, kandungan peroksida akan habis sehingga reaksi berubah menjadi  $CuCl_2 + Cu \longrightarrow 2CuCl$ . Cairan pelarut akan berubah menjadi warna coklat dan tidak dapat melarutkan tembaga lagi.

Sebelum pencampuran bahan pelarut, perlu diperhatikan takaran dari masing-masing bahan. Takaran yang pas untuk pencampuran  $HCL$ ,  $H_2O_2$  dan  $H_2O$  ialah 1:2:4. Hal ini sangat penting diperhatikan karena apabila takaran tidak pas dapat menimbulkan reaksi yang tidak diinginkan ataupun kegagalan pada proses etching. Contohnya apabila kandungan  $HCL$  terlalu tinggi dapat merusak jalur PCB.

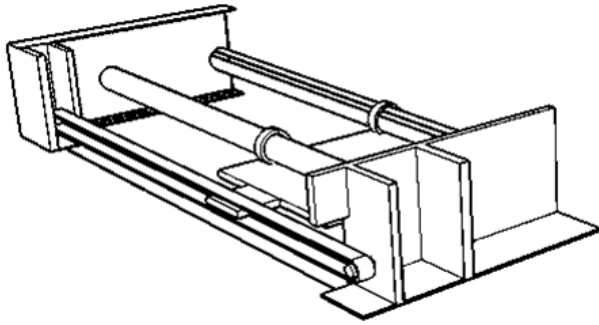
## 2.9 Slider sebagai Penggerak Lengan Robot

Dalam pembuatan mesin etching PCB otomatis, lengan robot dilengkapi *slider* yang menggerakkan lengan robot saat memindahkan PCB dari cairan  $NaOH$  menuju cairan  $HCL$ ,  $H_2O_2$  dan  $H_2O$  lalu menuju cairan air dan kemudian kembali lagi ke posisi awal pengambilan PCB. *Slider* tersebut menggunakan motor *stepper* karena sudut rotasi motor proporsional dengan pulsa masukan sehingga lebih mudah diatur, motor dapat langsung memberikan torsi penuh pada saat mulai bergerak, posisi dan pergerakan repetisinya dapat ditentukan secara presisi, dapat menghasilkan perputaran yang lambat sehingga beban dapat dikopel langsung ke porosnya, dan frekuensi perputaran dapat ditentukan secara bebas dan mudah pada range yang luas. Sehingga penggunaan motor *stepper* sangat tepat untuk menggerakkan lengan robot saat proses etching otomatis.

Motor *stepper* tidak dapat bergerak sendiri secara kontinyu, tetapi bergerak secara diskrit per-*step* sesuai dengan spesifikasinya. Untuk bergerak dari satu *step* ke *step* berikutnya diperlukan waktu dan menghasilkan torsi yang besar pada kecepatan rendah. Salah satu karakteristik Motor *Stepper* yang utama adalah mempunyai torsi penahan yang memungkinkan motor *stepper* menahan posisi yang berguna apabila membutuhkan keadaan *start* dan *stop*.

Motor *Stepper* yang digunakan untuk *slider* adalah motor *stepper bipolar* Nema 17. Motor *Stepper* Nema 17 merupakan jenis motor *Stepper Bipolar*. Motor *Stepper Bipolar* mempunyai 2 kutub yang aktif pada bagian stator apabila dialiri arus listrik. Motor ini mempunyai 4 kabel input. Motor *Stepper* Nema 17 mempunyai holding *torque* sebesar 4.8 kg yang berarti Motor ini dapat menahan beban pada saat keadaan berhenti maksimal 4.8 kg. Besar sudut per-*step* adalah 1.80 dengan akurasi 5%. Tegangan yang dibutuhkan oleh motor ini adalah 12V dengan arus 2.5 *Ampere*. Dengan berat lengan robot yang kurang dari 4.8 kg, jenis motor *stepper* yang di gunakan slider untuk menggerakkan lengan robot tergolong cukup kuat. Desain *slider* dapat dilihat pada Gambar 2.18.





**Gambar 2.18** Desain Slider Penggerak Lengan Robot

-----Halaman ini sengaja dikosongkan-----

## BAB III

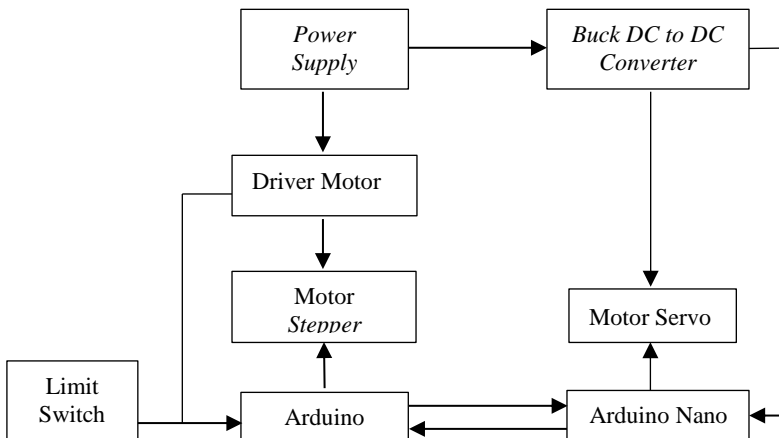
### PERANCANGAN SISTEM KONTROL

Pada bab ini akan dibahas mengenai perancangan perangkat keras dan lunak yang dilakukan dengan metode penelitian berdasarkan pada studi kepustakaan berupa data-data literatur dari masing-masing komponen, informasi dari *internet*, dan konsep-konsep teoritis dari buku-buku penunjang.

Perancangan diperlukan agar dalam tahapan selanjutnya berjalan dengan lancar, pada awalnya dilakukan perancangan perangkat keras. Setelah itu akan diuji dengan menggunakan perangkat lunak untuk memastikan keduanya dapat berjalan dengan baik.

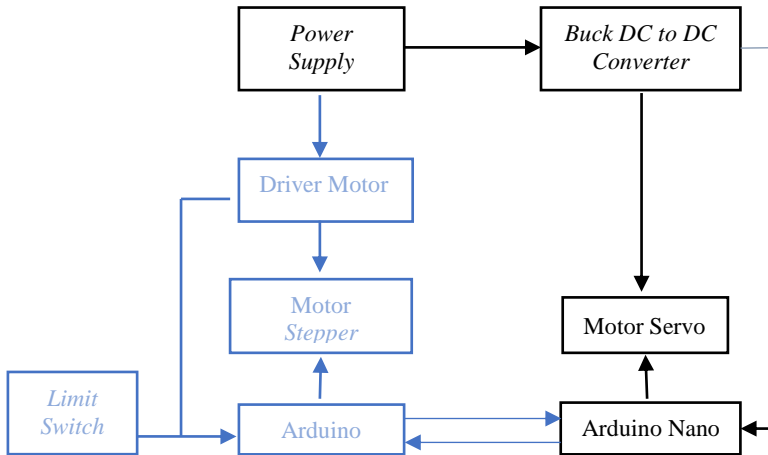
#### 3.1 Blok Fungsional Sistem

Sebelum melakukan perancangan perangkat keras dan perangkat lunak, diperlukan sebuah perancangan blok fungsional sistem berupa blok diagram yang menjelaskan sistem kerja secara keseluruhan Tugas Akhir ini. Secara keseluruhan blok fungsional sistem dapat dilihat pada Gambar 3.1 berikut.



**Gambar 3.1** Blok Diagram Fungsional Keseluruhan Sistem

Dalam pembuatan alat *etching* PCB otomatis, lengan robot dilengkapi *slider* sebagai penggerak sumbu x. Cara kerja keseluruhan sistem dapat dilihat pada Gambar 3.1. *Power Supply* merubah tegangan AC menjadi DC sebesar 12V 10 *ampere* kemudian memberikan tegangan tersebut kepada *driver* motor dan *driver* motor akan memberikan tegangan 5 volt kepada Arduino UNO. Arduino UNO mengontrol pergerakan motor stepper melalui *driver* motor. *Buck Converter* yang juga diberi tegangan oleh *power supply* merubah tegangan tersebut menjadi 6 Volt dan dimasukkan ke Arduino nano. Arduino ano mengatur pergerakan derajat motor servo. Arduino Nano dan Arduino Uno melakukan komunikasi melalui data *serial* sehingga kerja dari motor *stepper* dan motor servo saling bergantian. *Limit switch* bekerja sebagai tanda dari alat mulai bekerja.



**Gambar 3.2** Blok Diagram Fungsional Lengan Robot

Gambar 3.2 menjelaskan tentang urutan *Hardware* bekerja. Dapat dilihat warna biru merupakan bagian diagram fungsional *slider* penggerak lengan robot yang dikerjakan oleh A Dimas Maestrodia (2214030009) dan warna hitam merupakan diagram fungsional lengan robot yang dikerjakan oleh penulis. *Power Supply* mengubah daya dari PLN 220 V menjadi 12V dan di berikan kepada *DC to DC converter* untuk diubah sesuai tegangan yang diperlukan oleh Arduino dan motor servo yaitu

sebesar 6V. Arduino mengatur derajat tiap motor servo sesuai posisi yang diinginkan.

Untuk memudahkan dalam pembahasan bab ini akan dibagi menjadi dua bagian yaitu :

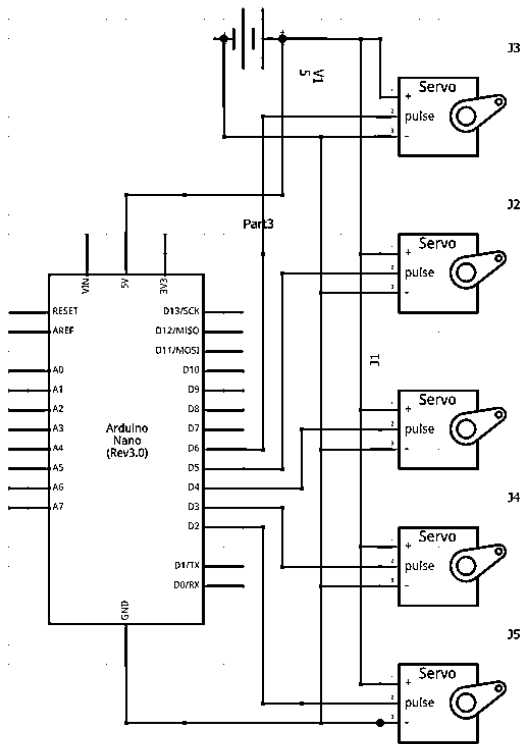
1. Perancangan *hardware* (perangkat keras) yang terdiri dari pembahasan pembuatan perangkat elektronik dan pembuatan perangkat mekanik.
2. Perancangan *software* (perangkat lunak) meliputi perancangan pembuatan flowchart program dan perancangan program untuk menjalankan motor servo.

### **3.2 Pembuatan Perangkat Elektronik**

Dalam perangkat elektronik, terdapat beberapa elemen yang harus disusun untuk dapat menggerakkan motor servo dengan baik. Elemen-elemen tersebut adalah sebagai berikut :

#### **3.2.1 Pengkabelan Arduino Nano Dengan Motor Servo**

Motor servo berfungsi sebagai aktuator untuk menggerakkan lengan robot. Semua pin PWM dihubungkan dengan pin PWM pada motor servo. Pin PWM dari pengendali ke motor servo berfungsi untuk mengirimkan sinyal PWM agar motor servo dapat bergerak sesuai dengan program. Dapat dilihat pada Gambar 3.3 dari atas ke bawah yaitu motor servo penjepit, pemutar, lengan atas, lengan bawah, dan basis.

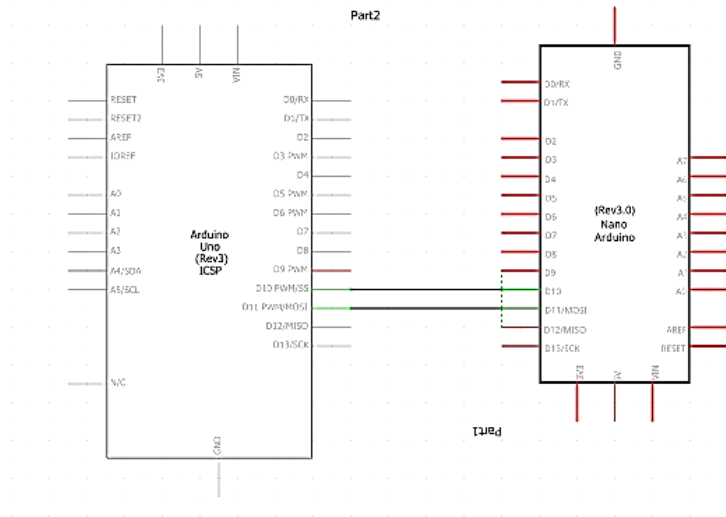


**Gambar 3.3** Pengkabelan Arduino Nano Dengan Motor Servo.

### 3.2.2 Rangkaian Komunikasi Data Serial Antara Arduino

Arduino merupakan sebuah mikrokontroler yang dapat digunakan untuk banyak proses kontrol sesuai dengan yang diinginkan oleh pengguna. Salah satu fungsi dari arduino adalah dapat bertukar data antar perangkat satu dengan yang lainnya. Komunikasi antar perangkat tersebut dapat terjadi secara parallel dan secara seri. Kelebihan dari komunikasi parallel adalah pengiriman data yang lebih cepat karena data yang akan dikirimkan dikirim bersamaan. Namun, komunikasi secara parallel ini akan membutuhkan banyak port pada arduino sesuai dengan banyak data yang akan dikirim. Apabila komunikasi secara seri hanya membutuhkan port yang sedikit, namun membutuhkan waktu yang lebih lama dibandingkan

dengan komunikasi secara parallel. Maka dari itu, komunikasi secara seri akan lebih efisien daripada parallel yang membutuhkan banyak port. Komunikasi serial yang dipakai pada arduino adalah pada pin 10 dan 11. Yang berfungsi sebagai TX atau *transfer* atau RX atau *receiver*. Komunikasi antar arduino seperti yang terlihat pada Gambar 3.4.



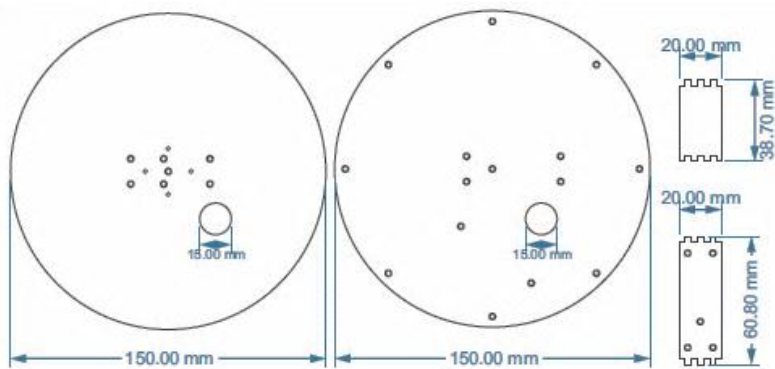
**Gambar 3.4** Rangkaian Komunikasi Data Antar Arduino

### 3.3 Pembuatan Perangkat Mekanik

Dalam perangkat mekanik, terdapat beberapa bagian yang harus dibuat untuk dapat bekerja dengan baik. Terdapat 4 bagian dalam perangkat mekaniknya yang terdiri dari dasar, lengan bawah, lengan atas, dan *end-effector* yang berupa penjepit. Berikut tahapan-tahapan pembuatan perangkat mekaniknya :

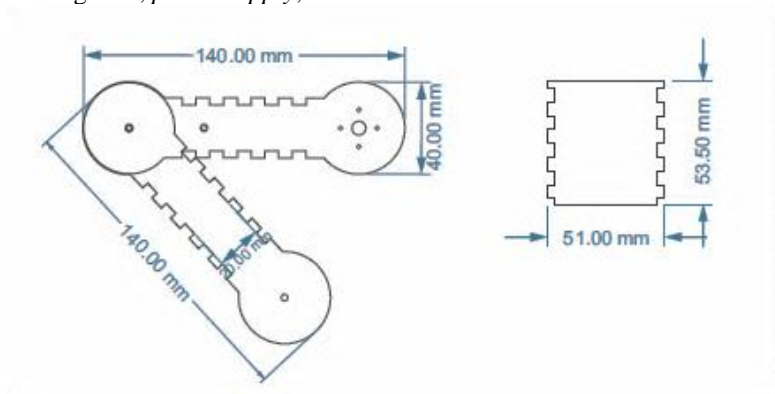
#### 3.3.1 Pembuatan Lengan Robot

Lengan robot terdiri dari 4 bagian yaitu bagian dasar, lengan bawah, lengan atas, dan *end-effector* yang berupa penjepit. Semua bahan yang digunakan untuk membuat lengan robot ini terbuat dari bahan akrilik. Sedangkan untuk penghubung digunakan mur dan baut berdiameter 3 mm.



**Gambar 3.5** Rancangan Bagian Dasar Lengan Robot

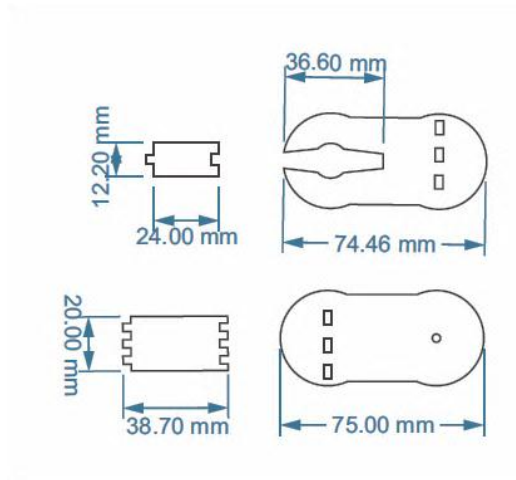
Gambar 3.5 adalah rancangan bagian dasar lengan robot yang berfungsi sebagai penghubung antara lengan bawah dengan tanah. Selain itu, pada bagian dasar berfungsi sebagaiudukan sistem minimum ATmega328, power supply, dan Buck DC to DC Converter.



**Gambar 3.6** Rancangan Bagian Lengan Bawah Robot

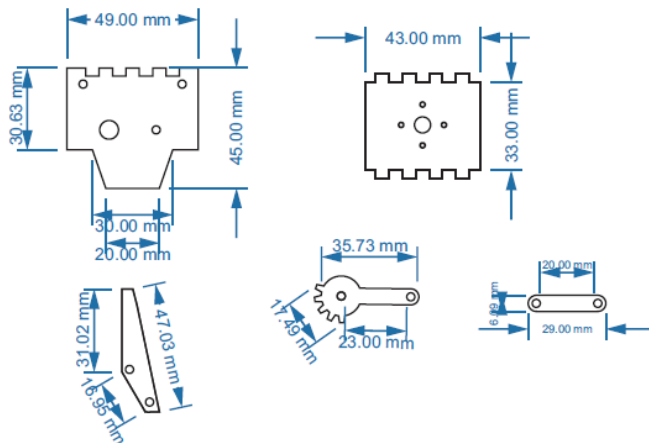
Pada Gambar 3.6 terdapat bagian yang secara langsung terhubung dengan motor servo dof 1 dan dof 2. Terdapat 2 link dikarenakan ukuran komponen tiap link yang sama. Bagian ini berfungsi sebagai link 1 dan link 2 atau penghubung antara basis dengan link 2 robot dan link 2 dengan link 3 robot.





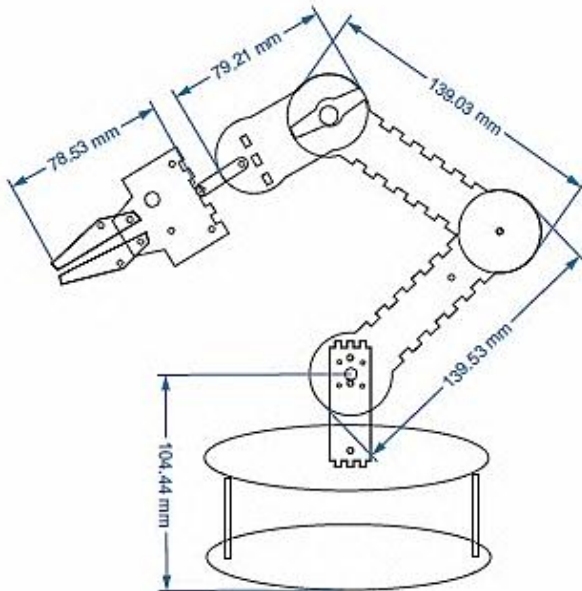
**Gambar 3.7** Rancangan Bagian Lengan Atas Robot

Gambar 3.7 adalah bagian lengan atas robot yang secara langsung terhubung oleh motor servo dof 3. Dihubungkan dengan sebuah lempeng akrilik yang menempel pada gear motor servo dof 3. Bagian ini berfungsi sebagai *link 3* atau penghubung antara *link 2* dengan *end-effector* robot.



**Gambar 3.8** Bagian Peralatan Lengan Robot

Gambar 3.8 merupakan bagian yang berupa ujung lengan robot yang berfungsi sebagaiudukan peralatan lengan robot. Dalam hal ini yang dimaksud dengan peralatan adalah pemutar dan penjepit yang digerakkan oleh dua buah motor mikro servo. Berbeda dengan gerakan servo-servo sebelumnya. Mikro servo yang terletak pada bagian peralatan ini hanya memutar, membuka dan menutup gripper, sedangkan servo yang lain bergerak secara rotasi antara sudut  $0^{\circ}$ - $180^{\circ}$ . Hasil penggabungan keseluruhan bagian lengan robot dapat dilihat pada Gambar 3.9.



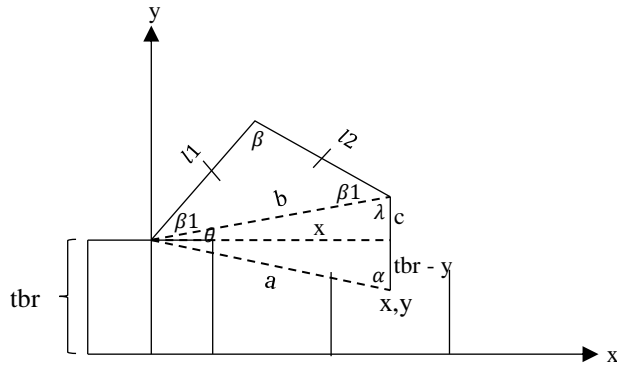
**Gambar 3.9** Hasil Rancangan Keseluruhan Bagian

### 3.4 Pembuatan Perangkat Lunak

Pada lengan robot penjepit PCB terdapat beberapa program yang harus dibuat untuk dapat menggerakkan motor *servo* dengan baik. Dalam pembuatan program kita menggunakan Arduino IDE untuk membuat *software* (perangkat lunak) yang menunjang kinerja dari komponen *Hardware* (perangkat keras).

### 3.4.1 Perhitungan *Inverse Kinematic* Lengan Robot

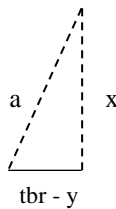
Untuk mendapatkan sudut masing-masing *joint* lengan robot penjepit PCB 3 DOF menggunakan perhitungan *inverse kinematic*, langkah pertama dengan memodelkan pergerakan robot seperti pada Gambar 3.10.



**Gambar 3.10** Pemodelan Kinematik Lengan Robot

Diketahui panjang  $l_1, l_2, l_3$  dan posisi  $x, y$  maka :

Ditarik garis dari base robot ke ujung lengan robot dan diberi nama variable  $a$ , nilai panjang  $a$  dapat dicari dengan rumus pythagoras seperti pada Gambar 3.11.



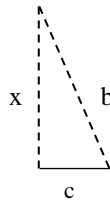
**Gambar 3.11** Mencari Panjang  $a$

$$a = \sqrt{(tbr - y)^2 + x^2}$$

Untuk mengetahui panjang alas segitiga sama kaki harus mencari nilai  $c$  terlebih dahulu.

$$c = l_3 + y - tbr$$

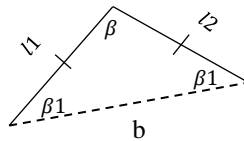
Nilai panjang dapat dicari menggunakan rumus pythagoras juga dari segitiga pada Gambar 3.12.



**Gambar 3.12** Mencari Panjang b

$$b = \sqrt{c^2 + x^2}$$

Selanjutnya yaitu mencari nilai sudut  $\beta$  pada segitiga sama kaki pada Gambar 3.13 yang juga merupakan nilai DOF 2 menggunakan aturan segitiga cosinus.



**Gambar 3.13** Mencari Nilai Sudut  $\beta$

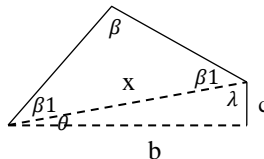
$$b^2 = l1^2 + l2^2 - 2l1 l2 \cos\beta$$

$$2 l1 l2 \cos\beta = l1^2 + l2^2 - b^2$$

$$\beta = \cos^{-1} \left( \frac{l1^2 + l2^2 - b^2}{2 l1 l2} \right)$$

Karena segitiga sama kaki, maka dengan mudah didapatkan nilai sudut  $\beta1$  yaitu :

$$\beta1 = \frac{180 - \beta}{2}$$



**Gambar 3.14** Mencari Nilai Sudut  $\theta$  Dan  $\lambda$

Untuk mencari nilai sudut  $\theta$  dan  $\lambda$  pada segitiga Gambar 3.14 dapat menggunakan rumus tan.

$$\theta = \tan^{-1} \frac{c}{x}$$

$$\lambda = \tan^{-1} \frac{c}{x}$$

Dari perhitungan di atas, didapatkan nilai sudut masing-masing *joint* yaitu :

$$dof1 = \beta1 + \theta$$

$$dof2 = \beta$$

$$dof3 = \beta1 + \lambda$$

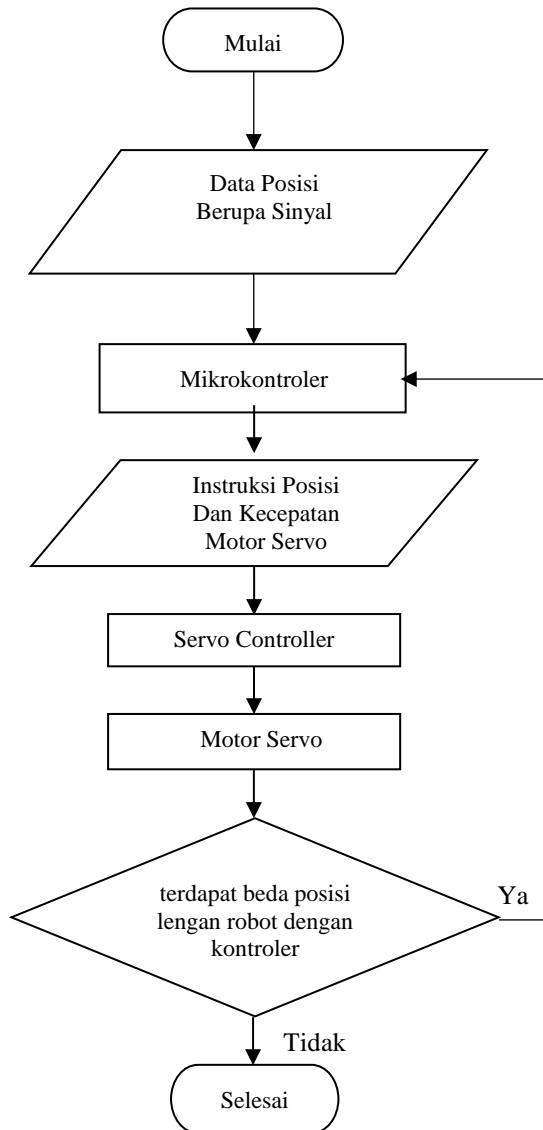
### 3.4.2 Pembuatan *Flowchart* Program

*Flowchart* merupakan bagan yang memperlihatkan urutan dan hubungan antar proses beserta instruksinya. Bagan ini dinyatakan dengan simbol. Dengan demikian setiap simbol menggambarkan proses tertentu. Sedangkan hubungan antar proses digambarkan dengan garis penghubung. *Flowchart* ini merupakan langkah awal pembuatan program. Dengan adanya *flowchart* urutan poses kegiatan menjadi lebih jelas.

Untuk pengolahan data dengan komputer, dapat dirangkum urutan dasar untuk pemecahan suatu masalah, yaitu;

1. *START*: berisi instruksi untuk persiapan peralatan yang diperlukan sebelum menangani pemecahan masalah.
2. *READ*: berisi instruksi untuk membaca data dari suatu peralatan.
3. *PROCESS*: berisi kegiatan yang berkaitan dengan pemecahan persoalan sesuai dengan data yang dibaca.
4. *WRITE*: berisi instruksi untuk merekam hasil kegiatan ke peralatan *output*.
5. *END*: mengakhiri kegiatan pengolahan

*Flowchart* program dari Tugas Akhir ini meliputi seluruh sistem jalannya alat ini yang dapat dilihat pada Gambar 3.15. Sistem yang dimaksud adalah sistem umum secara keseluruhan.



**Gambar 3.15** *Flowchart* Keseluruhan Program

### 3.4.3 Pendefinisian Pin dan Pustaka

```
#include <Servo.h>

Servo upper_arm, forearm, wrist, roll, gripper;

int s1 = 2;
int s2 = 3;
int s3 = 4;
int s4 = 5;
int s5 = 6;
```

**Gambar 3.16** Program Definisi Pustaka Dan Tipe Data

Dalam rancangan program yang dibuat, dimasukkan pustaka untuk kendali motor servo yang telah disediakan oleh Arduino IDE. Setelah memasukan pustaka servo, diberi nama setiap servo yang akan digunakan dalam pemrograman, dalam hal ini terdapat 5 servo yang masing-masing diberi nama sesuai bagian servo pada lengan robot yaitu *upper\_arm*, *forearm*, *wrist*, *roll*, dan *gripper*. Hasil pendefinisian pin dan pustaka dapat dilihat pada Gambar 3.16.

Kemudian, “int” digunakan untuk menyatakan tipe data *integer* (bilangan bulat) yang dapat memuat 16-bit dimulai dari -32,768 hingga 32,767. Nilai bilangan ini membutuhkan 2 byte memori mikrokontroller.

Definisi s1 pada segmen program tersebut untuk mendefinisikan sebuah variabel bernama s1 yang berada pada pin 2, s2 yang berada pada pin 3, s3 yang berada pada pin 4, s4 yang berada pada pin 5, dan s5 yang berada pada pin 6 Arduino.

### 3.4.4 Segmen Program Untuk Pengaturan Awal

```
void setup() {
    upper_arm.attach(s1);
    forearm.attach(s2);
    wrist.attach(s3);
    roll.attach(s4);
    gripper.attach(s5);
    Serial.begin(9600);
}
```

**Gambar 3.17** Segmen Program Untuk Pengaturan Awal

*Void* pada Arduino berfungsi sebagai kata kunci untuk membuat subprogram dari program utama. Jika fungsi *void* digunakan sebagai *void setup*, maka subprogram tersebut berfungsi sebagai program awal dari Arduino dan tidak diulang saat program dieksekusi.

Pada Gambar 3.17, terdapat fungsi *attach* yang hanya dapat dikenali apabila memasukan pustaka *servo* dalam program. Maksud dari fungsi *attach* adalah menyatakan bahwa *upper\_arm* dihubungkan dengan *s1* yang sudah didefinisikan sebelumnya yaitu terhubung dengan pin 2 Arduino. Dengan demikian, pada pin 2 akan menghasilkan gelombang PWM.

### 3.4.5 Segmen Program Untuk Menggerakkan Servo

```
void gripper_open() {  
    gripper.write(0);  
}  
void gripper_close() {  
    gripper.write(180);  
}  
  
void roll_over() {  
    roll.write(80);  
    delay(20);  
    roll.write(100);  
    delay(20);  
}
```

**Gambar 3.18** Segmen Program Untuk Menggerakkan Servo

Pada Gambar 3.18, terdapat fungsi *void* yang diikuti oleh nama subprogram yaitu *gripper\_open*, *gripper\_close* dan *roll\_over*. Dari masing-masing *sub program* terdapat fungsi *write* yang hanya dapat dikenali apabila memasukan pustaka *servo* dalam program.

Maksud dari fungsi *write* yang dihubungkan dengan nama motor servo adalah memanipulasi gelombang PWM agar motor servo dapat bergerak sebesar sudut yang dituliskan dalam konstanta dalam kurung.

### 3.4.6 Setting Program Komunikasi Serial Arduino

Program untuk komunikasi serial arduino merupakan sebuah program yang bisa menyelaraskan dua atau lebih arduino dengan urutan kerja sesuai dengan yang pengguna inginkan. Dalam program kali ini, program akan diselaraskan dengan program untuk menggerakkan lengan



robot. Sehingga slider dan lengan robot akan dapat bergerak secara bergantian secara otomatis.

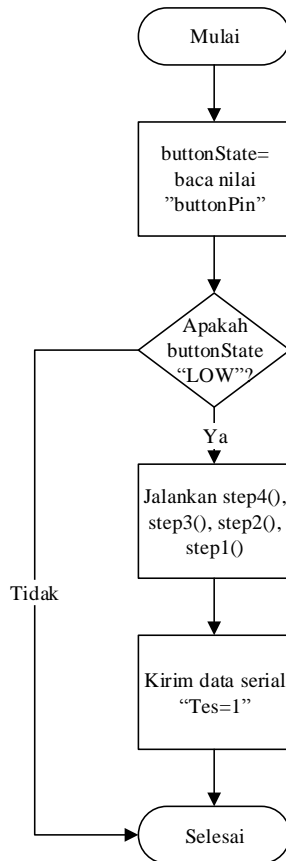
Urutan dari pergerakan *hardware* adalah pada saat lengan sudah menjepit pcb dengan benar, maka slider akan mulai bergerak untuk memposisikan lengan kedepan wadah yang berisi larutan NaOH yang akan melarutkan PCB setelah dilakukan penyinaran UV. Apabila sudah berada pada posisi yang tepat maka lengan selanjutnya akan bergerak untuk membawa PCB kedalam larutan NaOH. Setelah itu, maka PCB akan dibawa kedalam larutan  $HCL$ ,  $H_2O_2$  dan  $H_2O$  untuk dilakukan proses etching PCB selama beberapa menit. Setelah tembaga dari PCB hilang dengan proses etching maka akan dibersihkan dengan air bersih untuk selanjutnya dilakukan proses pengeboran.

Pin yang digunakan adalah pin 10 dan pin 11 yang berfungsi sebagai Transmitter dan Receiver. Pada program digunakan library software serial untuk supaya dapat mengakses pin secara serial. Baud rate yang digunakan adalah 11500 pada kedua program yaitu untuk transmitter dan untuk *receiver*.

Program yang berfungsi untuk memberikan sinyal kepada arduino lain bahwa tugas telah selesai dikerjakan dapat diperlihatkan seperti Gambar 3.19. pada program tersebut, perintah yang digunakan sebagai pengirim sinyal adalah “*serial.println(“”)*”. Program tersebut dapat diakses dengan library “*softwareserial.h*”. Flowchart program dapat dilihat pada Gambar 3.20.

```
#include <SoftwareSerial.h>
int buttonState = 0;
void setup() {
  pinMode(buttonPin, INPUT_PULLUP);
  Serial.begin(115200);
}
void loop() {
  buttonState = digitalRead(buttonPin);
  if (buttonState == LOW) {
    step4();
    step3();
    step2();
    step1();
    Serial.println("signal transfer");
  }
}
```

**Gambar 3.19** Program Transfer Komunikasi Serial



**Gambar 3.20** *Flowchart* Pengiriman Data Komunikasi Serial

Sedangkan program yang digunakan untuk penerimaan sinyal apabila tugas telah diselesaikan oleh arduino lain, ditunjukkan pada Gambar 3.21.

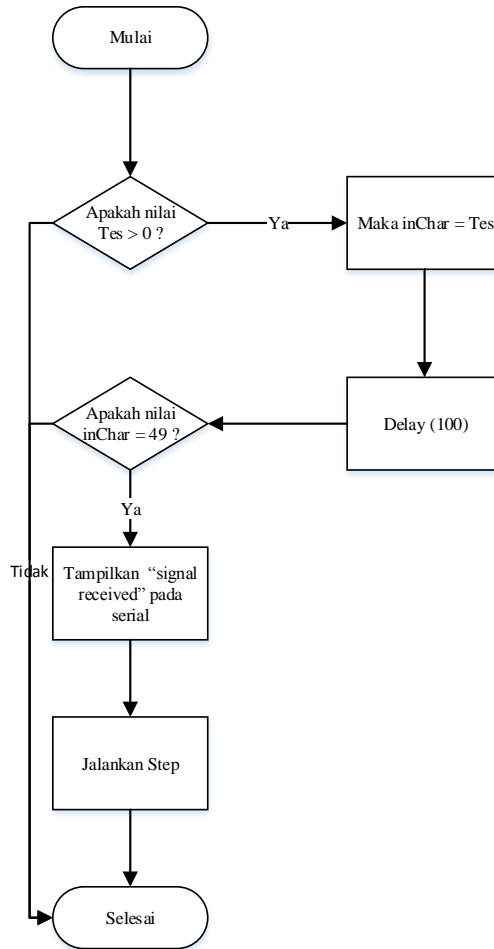
```

#include <SoftwareSerial.h>
int buttonState = 0;
void setup() {
    pinMode(buttonPin, INPUT_PULLUP);
    Serial.begin(115200);
}
void loop() {
    // put your main code here, to run repeatedly:
    if(tes.available()>0){
        inChar = tes.read();
    }
    //Serial.println(inChar);
    delay(100);
    if(inChar==49){
        Serial.println("signal received");
        delay(1000);
        while(flag==1){
            for(i=0;i<80;i++){
                step1();
            }
            delay(100);
        }
    }
}

```

**Gambar 3.21** Program *Receiver* Komunikasi Serial

Perintah diatas yang digunakan untuk penerimaan sinyal adalah `inChar = tes.read()`". Sedangkan alur kerja dari program penerimaan sinyal data dapat dilihat pada gambar 3.22.



**Gambar 3.22** Flowchart Penerimaan Data Komunikasi Serial

Alur kerja dari Penerimaan data berawal dari pengecekan nilai tes. Apabila nilai tes lebih besar dari nol, maka nilai *inChar* sama dengan nilai tes yang telah dikirimkan pada program yang sebelumnya. Selanjutnya, jika nilai *inChar* adalah 49 dalam kode ASCII atau dalam kode decimal adalah satu, maka program akan menuliskan secara serial “signal received”, lalu akan menjalankan step 1 pada motor stepper.

## BAB IV

### HASIL SIMULASI DAN IMPLEMENTASI

#### 4.1 Data Pengukuran *Switch-Mode Power Supply*

*Power Supply* yang digunakan mempunyai spesifikasi tegangan masukan maksimal 110/240 Volt dengan toleransi 15%. Sedangkan keluaran dari *power supply* berdasarkan spesifikasi sebesar 12 Volt dan 10 *ampere*. Pada Tabel 4.1 diperlihatkan besar tegangan output dari *power supply* yang telah diukur menggunakan *avometer*.

**Tabel 4.1** Data Pengukuran *Power Suplly*

Keterangan	Tegangan yang Dibutuhkan (V)	Tegangan <i>Output</i> (V)
Tanpa Beban	11-12	11.48
Dengan beban <i>slider</i>	11-12	11.48
Dengan beban lengan robot	11-12	11.47
Dengan beban keduanya	11-12	11.46

Dari hasil pengukuran pada Tabel 4.1 dapat diketahui bahwa tegangan *output* pada *power supply* hanya sekitar 11.46 sampai dengan 11.48 Volt tetapi masih dalam *range* tegangan yang dibutuhkan alat.

#### 4.2 Data Pengukuran *Buck Converter DC to DC*

*Buck converter* yang digunakan yaitu *buck converter DC to DC* tipe XL4005 yang mempunyai spesifikasi tegangan masukan maksimal 4-35 Volt, tegangan keluaran 1.25-32 Volt dan arus sebesar 5 *ampere*. *Buck converter DC to DC* digunakan sebagai penurun tegangan sesuai yang diinginkan. Cara menurunkan tegangan yaitu dengan memutar potensio. Tegangan input yang diterima dari *power supply* yaitu sebesar 12 V. Cara pengambilan data yaitu dengan mengukur tegangan output *buck converter* pada saat tanpa beban dan dengan beban. Setelah itu dapat dilihat apakah nilai tegangan output sesuai dengan yang diperlukan alat. Data pengukuran tegangan ditunjukkan pada Tabel 4.2.

**Tabel 4.2** Data Pengukuran *Buck Converter*

Keterangan	Tegangan yang Diperlukan (V)	Tegangan <i>Output</i> (V)
Tanpa Beban	5 V	5.01 V
Dengan Beban	5 V	4.99 V

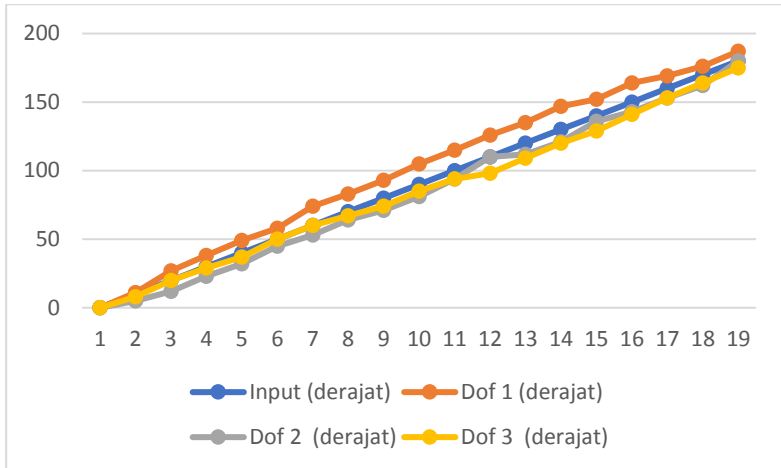
Dari hasil pengambilan data *Buck Converter* didapatkan tegangan *output* saat tanpa beban sebesar 5.01 V dan tegangan *output* saat ada beban sebesar 4.99 V. Hal tersebut sudah sangat mendekati nilai tegangan yang diperlukan alat.

### 4.3 Pengukuran Dan Pengujian Arduino Nano Dengan Motor *Servo*

Tujuan dari pengambilan data adalah untuk mengetahui kesalahan sudut yang dihasilkan dari masing-masing motor *servo*. Untuk melakukan pengukuran gerakan motor *servo*, maka dilakukan pemrograman melalui Arduino dengan *interval* sudut sebesar 10° hingga 180° dan diukur tiap motor *servo* dengan busur. Data pengukuran *Output servo* dapat dilihat pada Tabel 4.3 dan data dalam grafik dapat dilihat pada Gambar 4.1.

**Tabel 4.3** Data Pengukuran *Output Servo*

No.	Input (derajat)	Dof 1 (derajat)	Dof 2 (derajat)	Dof 3 (derajat)
1.	0	0	0	0
2.	10	11	5	8
3.	20	27	12	20
4.	30	38	23	29
5.	40	49	32	37
6.	50	58	45	50
7.	60	74	53	60
8.	70	83	64	67
9.	80	93	71	74
10.	90	105	81	85
11.	100	115	94	94
12.	110	126	110	98
13.	120	135	112	109
14.	130	147	121	120
15.	140	152	136	129
16.	150	164	143	141
17.	160	169	153	153
18.	170	176	162	164
19.	180	187	180	175



**Gambar 4.1** Grafik Data Pengukuran *Output* Servo

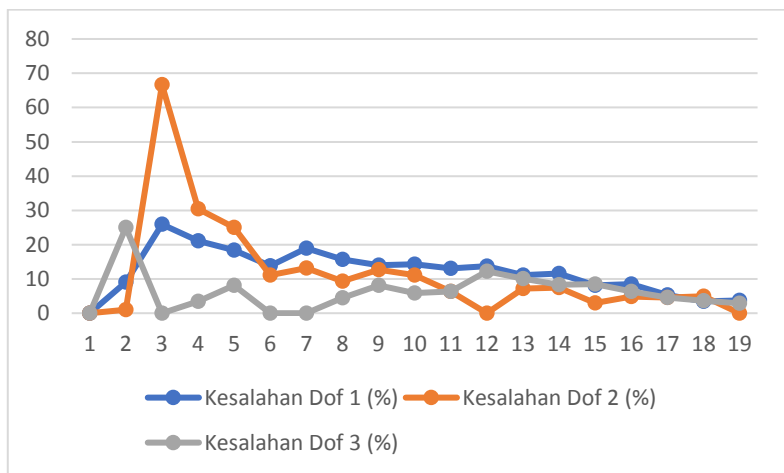
Pada percobaan yang dilakukan, didapat bahwa masing-masing motor *servo* memiliki kesalahan dalam menghasilkan sudut. Data presentase kesalahan sudut servo dapat dilihat pada Tabel 4.4 dan grafik kesalahan sudut servo dappat dilihat pada Gambar 4.2. Untuk mengetahui persen kesalahan sudut tiap servo dapat dihitung menggunakan rumus :

$$\text{Persen Kesalahan} = \frac{|\text{Derajat output servo} - \text{input derajat}|}{\text{Derajat output servo}} \times 100$$

**Tabel 4.4** Persentase Kesalahan Sudut Servo

No.	Kesalahan Dof 1 (%)	Kesalahan Dof 2 (%)	Kesalahan Dof 3 (%)
1.	0	0	0
2.	9.09090909	1	25
3.	25.9259259	66.6666667	0
4.	21.0526316	30.4347826	3.44827586
5.	18.3673469	25	8.10810811
6.	13.7931034	11.1111111	0
7.	18.9189189	13.2075472	0
8.	15.6626506	9.375	4.47761194
9.	13.9784946	12.6760563	8.10810811
10.	14.2857143	11.1111111	5.88235294

11.	13.0434783	6.38297872	6.38297872
12.	13.6984127	0	12.244898
13.	11.11111111	7.14285714	10.0917431
14.	11.5646259	7.43801653	8.33333333
15.	8.16326531	2.94117647	8.52713178
16.	8.53658537	4.8951049	6.38297872
17.	5.32544379	4.5751634	4.5751634
18.	3.40909091	4.9382716	3.65853659
19.	3.74331551	0	2.85714286



**Gambar 4.2** Grafik Presentase Kesalahan Sudut Servo

#### 4.4 Pengujian Daya Cengkram Gripper

Pengujian dilakukan dengan cara mengambil satu persatu PCB berbagai ukuran dan memutarnya dengan kecepatan sedang dan waktu maksimal 5 menit karena lama proses etching dengan lengan robot membutuhkan waktu sekitar 5 menit. Pada tiap ukuran PCB akan dicatat seberapa lama gripper dapat mencengkram PCB dengan ukuran tertentu. Hasil pengamatan dicatat pada Tabel 4.5.

**Tabel 4.5** Data Pengamatan Daya Cengkram Gripper

No.	Dimensi PCB	Lama Putaran	Hasil Pengamatan
1.	3 x 5	5 menit	Tidak Terlepas
2.	5 x 4	5 menit	Tidak Terlepas



3.	5 x 6	5 menit	Tidak Terlepas
4.	7 x 4	5 menit	Tidak Terlepas
5.	8 x 4	5 menit	Tidak Terlepas
6.	4 x 9	4 menit 27 detik	Terlepas
7.	5 x 9	4 menit 20 detik	Terlepas
8.	6 x 7	3 menit 50 detik	Terlepas
9.	7 x 7	3 menit 32 detik	Terlepas
10.	6 x 10	2 menit 54 detik	Terlepas

Berdasarkan pengujian yang dilakukan, terdapat dua keadaan yang dialami PCB saat dicengkram *gripper* yaitu terlepas dan tidak terlepas. PCB dengan ukuran kecil dan sedang tidak akan terlepas meskipun diputar selama 5 menit. Sedangkan PCB berukuran besar sudah tidak dapat diatasi oleh *gripper* pada menit tertentu. Gambar saat pengambilan data dapat dilihat pada Gambar 4.3.



**Gambar 4.3** Pengujian Daya Cengkram *Gripper*

#### **4.5 Pengukuran Kecepatan Putaran Gripper**

Pengujian dilakukan dengan cara mengambil satu PCB berukuran 8 x 4 dan memutarkannya dengan kecepatan berbeda-beda dan waktu maksimal

5 menit karena lama proses etching dengan lengan robot membutuhkan waktu sekitar 5 menit. Pada tiap ukuran PCB akan dicatat seberapa lama gripper dapat mencengkram PCB dengan kecepatan tertentu. Hasil pengamatan dicatat pada Tabel 4.6.

**Tabel 4.6** Data Pengamatan Kecepatan Putaran Gripper

No.	Kecepatan (delay/ms)	Lama Putaran	Hasil Pengamatan
1.	50	1 menit 54 detik	Terlepas
2.	75	2 menit 38 detik	Terlepas
3.	100	3 menit 31 detik	Terlepas
4.	125	4 menit 24 detik	Terlepas
5.	150	5 menit	Tidak Terlepas
6.	175	5 menit	Tidak Terlepas
7.	200	5 menit	Tidak Terlepas
8.	225	5 menit	Tidak Terlepas
9.	250	5 menit	Tidak Terlepas
10.	275	5 menit	Tidak Terlepas

Berdasarkan pengujian yang dilakukan, Pada kecepatan putaran dengan delay antara 50 - 125 PCB akan terlepas pada menit tertentu. Sedangkan PCB yang kecepatan putaran dengan delay 150 – 275 tidak terlepas sampai menit ke 5.

#### 4.6 Pengukuran Dan Pengujian Lama Proses *Etching* Pada Berbagai Ukuran PCB

Pengujian dilakukan dengan cara mengambil satu persatu PCB berbagai ukuran dan memutarkannya dalam cairan  $HCL$ ,  $H_2O_2$  dan  $H_2O$  hingga tembaga pada PCB larut. Pada tiap ukuran PCB akan dicatat seberapa lama waktu yang dibutuhkan untuk melarutkan tembaga secara sempurna. Hasil pengamatan dicatat pada Tabel 4.7.

**Tabel 4.7** Lama Proses Etching Pada Berbagai Ukuran PCB

No.	Dimensi PCB	Lama Putaran
1.	3 x 5	1 menit 20 detik
2.	5 x 4	1 menit 24 detik
3.	5 x 6	1 menit 23 detik
4.	7 x 4	1 menit 30 detik
5.	8 x 4	1 menit 30 detik

6.	4 x 9	1 menit 32 detik
7.	5 x 9	1 menit 31 detik
8.	6 x 7	1 menit 29 detik
9.	7 x 7	1 menit 28 detik
10.	6 x 10	1 menit 32 detik

Dari hasil pengujian dapat dilihat bahwa proses *etching* menggunakan cairan pelarut H3 yang terdiri dari  $HCL$ ,  $H_2O_2$  dan  $H_2O$  tergolong sangat cepat. Terdapat sedikit perbedaan waktu dalam proses etching pada tiap PCB yang berbeda ukuran. Proses etching PCB dengan ukuran kecil lebih cepat disbanding dengan PCB yang ukurannya besar

#### 4.7 Pengukuran dan Pengujian Kecepatan Putaran PCB Terhadap Lama Pelarutan Tembaga

Pengujian dilakukan dengan cara mengambil satu PCB berukuran 8 x 4 dan memutarinya dalam cairan  $HCL$ ,  $H_2O_2$  dan  $H_2O$  dengan berbagai kecepatan hingga tembaga pada PCB larut. Pada tiap kecepatan putaran motor akan dicatat hasil etching PCB dengan lama waktu 30 detik yang dibutuhkan untuk melarutkan tembaga secara sempurna. Hasil pengamatan dicatat pada Tabel 4.8.

**Tabel 4.8** Kecepatan Putaran PCB Terhadap Lama Pelarutan Tembaga

No.	Kecepatan (delay/ms)	Hasil Pengamatan
1.	50	Jalur PCB larut
2.	75	Jalur PCB larut
3.	100	Jalur PCB terputus-putus
4.	125	Jalur PCB terputus-putus
5.	150	Jalur tercetak sempurna
6.	175	Jalur tercetak sempurna
7.	200	Jalur tercetak sempurna
8.	225	Jalur belum terbentuk
9.	250	Jalur belum terbentuk
10.	275	Jalur belum terbentuk

Dari hasil pengujian dapat dilihat bahwa jalur PCB tercetak sempurna dengan *delay* putaran *grripper* antara 150 sampai dengan 200 atau dengan kecepatan sedang.

#### 4.8 Pengujian Keseluruhan Sistem

Pengujian keseluruhan sistem ini bertujuan untuk mengetahui ketepatan proses etching PCB mulai dari penjepitan PCB, pencengkaman PCB dan pelarutan tembaga. Gambar pengujian keseluruhan sistem dapat dilihat pada Lampiran B. Dari beberapa pengujian yang telah dilakukan pada lengan robot, PCB yang digunakan untuk proses etching dengan alat ini tidak lebih besar dari 8x4 cm, delay putaran gripper sebesar 150, dan pencelupan PCB pada campuran larutan  $HCL$ ,  $H_2O_2$  dan  $H_2O$  selama 30 detik. Pengambilan data dilakukan dengan melakukan proses etching sebanyak tiga kali dengan ukuran PCB yang berbeda. Terdapat lima tahap berdasarkan pengujian yang dilakukan diantaranya :

Tahap 1 : Pengambilan PCB di posisi awal

Tahap 2 : Pemindahan PCB dari posisi awal ke kotak yang berisi cairan NaOH

Tahap 3 : Pemindahan PCB dari kotak yang berisi cairan NaOH ke kotak yang berisi cairan  $HCL + H_2O_2 + H_2O$

Tahap 5 : Pemindahan PCB dari kotak yang berisi cairan  $HCL + H_2O_2 + H_2O$  ke kotak yang berisi air bersih

Tahap 6 : Pemindahan PCB dari kotak yang berisi air bersih ke tempat lain untuk dilihat hasil etching

Hasil pengamatan dapat dilihat pada Tabel 4.9 sampai dengan Tabel 4.18

**Tabel 4.9** Hasil Pengujian Dengan PCB Ukuran 3 x 5

Tahap 1	Tahap 2	Tahap 3	Tahap 4	Tahap 5
PCB Terambil	PCB berhasil dipindahkan ke NaOH	PCB berhasil dipindahkan ke $HCL, H_2O_2 + H_2O$	PCB berhasil dipindahkan ke air	Jalur PCB terputus-putus

**Tabel 4.10** Hasil Pengujian Dengan PCB Ukuran 5 x 4

Tahap 1	Tahap 2	Tahap 3	Tahap 4	Tahap 5
PCB Terambil	PCB berhasil dipindahkan ke NaOH	PCB berhasil dipindahkan ke $HCL, H_2O_2 + H_2O$	PCB berhasil dipindahkan ke air	Jalur PCB terputus-putus

**Tabel 4.11** Hasil Pengujian Dengan PCB Ukuran 5 x 6

Tahap 1	Tahap 2	Tahap 3	Tahap 4	Tahap 5
PCB Terambil	PCB berhasil dipindahkan ke NaOH	PCB berhasil dipindahkan ke $HCL, H_2O_2 + H_2O$	PCB berhasil dipindahkan ke air	Jalur PCB tercetak sempurna

**Tabel 4.12** Hasil Pengujian Dengan PCB Ukuran 7 x 4

Tahap 1	Tahap 2	Tahap 3	Tahap 4	Tahap 5
PCB Terambil	PCB berhasil dipindahkan ke NaOH	PCB berhasil dipindahkan ke $HCL, H_2O_2 + H_2O$	PCB berhasil dipindahkan ke air, cengkaman akan terlepas	Jalur PCB tercetak sempurna

**Tabel 4.13** Hasil Pengujian Dengan PCB Ukuran 8 x 4

Tahap 1	Tahap 2	Tahap 3	Tahap 4	Tahap 5
PCB Terambil	PCB berhasil dipindahkan ke NaOH	PCB berhasil dipindahkan ke $HCL, H_2O_2 + H_2O$	PCB terjatuh sebelum masuk ke kotak air	Jalur PCB tercetak sempurna

**Tabel 4.14** Hasil Pengujian Dengan PCB Ukuran 4 x 9

Tahap 1	Tahap 2	Tahap 3	Tahap 4	Tahap 5
PCB Terambil	PCB berhasil dipindahkan ke NaOH	PCB terlepas saat proses etching	-	Jalur PCB tercetak sempurna

**Tabel 4.15** Hasil Pengujian Dengan PCB Ukuran 5 x 9

Tahap 1	Tahap 2	Tahap 3	Tahap 4	Tahap 5
PCB Terambil	PCB terjatuh di kotak cairan NaOH	-	-	-

**Tabel 4.16** Hasil Pengujian Dengan PCB Ukuran 6 x 7

Tahap 1	Tahap 2	Tahap 3	Tahap 4	Tahap 5
PCB Terambil	PCB berhasil dipindahkan ke NaOH	PCB terlepas saat proses etching	-	-

**Tabel 4.17** Hasil Pengujian Dengan PCB Ukuran 7 X 7

Tahap 1	Tahap 2	Tahap 3	Tahap 4	Tahap 5
PCB Terambil	PCB berhasil dipindahkan ke NaOH	PCB terlepas saat proses etching	-	-

**Tabel 4.18** Hasil Pengujian Dengan PCB Ukuran 6 x 10

Tahap 1	Tahap 2	Tahap 3	Tahap 4	Tahap 5
PCB Terambil	PCB terjatuh di kotak cairan NaOH	-	-	-

Berdasarkan hasil pengujian keseluruhan sistem, terdapat kegagalan pada pemindahan PCB. PCB dengan ukuran 4x9, 6x7, dan 7x7 terlepas dari cengkaman *gripper* saat diputar di cairan  $HCL + H_2O_2 + H_2O$ . Hal itu dikarenakan pada kotak yang berisi cairan  $HCL + H_2O_2 + H_2O$  PCB diputar dengan waktu yang sedikit lama untuk melarutkan tembaga sehingga PCB mendapat tekanan dari cairan tersebut yang menyebabkan terlepasnya PCB dari cengkaman *gripper*. Sedangkan PCB dengan ukuran 5x9 dan 6x10 terlepas pada tahap 2 yaitu di kotak cairan NaOH dikarenakan dimensi yang terlalu besar untuk *gripper*. Selain itu, lengan robot hanya dapat melakukan etching selama empat kali berturut-turut dan setelah etching yang keempat posisi lengan robot berubah sehingga tidak dapat langsung digunakan lagi dan memerlukan waktu untuk dapat digunakan lagi. Gambar hasil etching PCB menggunakan alat ini dapat dilihat pada Lampiran B.

-----Halaman ini sengaja dikosongkan-----



## **BAB V**

### **PENUTUP**

Dari hasil yang telah didapatkan selama proses perancangan dan pembuatan serta proses pengujian dan analisa data untuk Tugas Akhir ini, maka dapat diambil beberapa kesimpulan dan saran yang berguna untuk perbaikan dan pengembangan agar nantinya bermanfaat.

#### **5.1 Kesimpulan**

1. Dari pengujian daya cengkram *gripper*, *gripper* tidak cukup kuat untuk mencekram PCB dengan ukuran besar. Ukuran PCB yang mampu dibawa lengan robot pada proses *etching* dengan alat ini yaitu 5x6, 7x4, dan 8x4 dan banyak PCB yang dapat di *etching* adalah empat buah dan setelah itu harus diberikan waktu untuk dapat digunakan lagi.
2. Waktu yang dibutuhkan oleh alat untuk melakukan *etching* PCB dengan beberapa ukuran rata-rata adalah satu menit dan tiga puluh detik.
3. Dari hasil pengambilan data output motor *servo*, didapat bahwa masing-masing *servo* memiliki kesalahan dalam menghasilkan sudut.

#### **5.2 Saran**

Saran untuk penelitian selanjutnya adalah sebaiknya digunakan motor dengan torsi yang lebih besar pada *gripper* agar dapat membawa PCB dengan ukuran lebih besar. Selain itu *gripper* dibuat lebih panjang agar sekrup tidak terkena cairan kimia untuk *etching* yang dapat menyebabkabn sekrup menjadi berkarat.

-----Halaman ini sengaja dikosongkan-----

## DAFTAR PUSTAKA

- [1] Anonym. “Otomasi Industri Buka Pintu Pabrik Pabrik Pintar Di Indonesia”. <http://www.suara.com/tekno/2016/06/01/095450/otomasi-industri-buka-pintu-pabrik-pabrik-pintar-di-indonesia>. diakses pada 8 Mei 2017.
- [2] Budiharto, Widodo 2009 "Membuat Sendiri Robot Cerdas (Edisi Revisi)". Jakarta: Elex Media Komputindo.
- [3] Anonym. “Servo Motor Tower Pro MG995 15Kg”. <http://www.coldfire-electronica.com/esp/item/26/servomotor-towerpro-mg995-15kg-cm>. diakses pada 09 Januari 2017.
- [4] Anonym. “TowerPro SG90 Servo Specification and Reviews”. <https://servodatabase.com/servo/towerpro/sg90>. diakses pada 12 April 2017.
- [5] Erwantono, Hendri., dan Safarina, Evi Siska. "Rancang Bangun Lengan Robot Berbasis ATmega328 Untuk Memindahkan Biskuit", *Tugas Akhir*, Institut Teknologi Sepuluh Nopember, Surabaya, 2016.
- [6] Servo City, “How Does a Servo Works”. [http://www.servocity.com/html/how\\_do\\_servos\\_work\\_.html](http://www.servocity.com/html/how_do_servos_work_.html). diakses pada 20 Februari 2017.
- [7] Anonym. “Standard Servo”. <http://learn.parallax.com/tutorials/language/propeller-c/propeller-c-simple-devices/standard-servo>. diakses pada 20 Mei 2017.
- [8] Anonym. “Mengenal SMPS Switched Mode Power Supply”. <https://fariztfarizt.wordpress.com/2013/12/27/mengenal-smps-switched-mode-power-supply/>. diakses pada 24 Mei 2017.
- [9] Ebay. “XL4005-DC 5A Adjustable Step Down Module Arduino Converter Power Supply”. <http://www.ebay.co.uk/itm/XL4005-DC-5A-Adjustable-Step-Down-Module-Arduino-Converter-Power-Supply-/181862201051>. diakses pada 12 April 2017.
- [10] Pitowarno, Endra 2006 “Robotika : Desain, Kontrol, Dan Kecerdasan Buatan”. Yogyakarta : Andi Offset

-----Halaman ini sengaja dikosongkan-----

## LAMPIRAN A

### A.1 Lampiran Program Arduino

```
#include <Servo.h>
#include <SoftwareSerial.h>
Servo upper_arm,forearm,wrist,roll,gripper;
#define speed_delay 150
int inChar;

const byte rxPin = 10;
const byte txPin = 11;
SoftwareSerial tes(rxPin, txPin);

int def_dof1=0;
int def_dof2=90;
int def_dof3=0;
int def_dof4=180;
int s1=2;
int s2=3;
int s3=4;
int s4=5;
int s5=6;
int min_dof1=170,min_dof2=80,min_dof3=20,min_dof4=10;
int max_dof1=0,max_dof2=180,max_dof3=90,max_dof4=165;
int i,sel1,sel2,sel3,sel4,min_s;
int prev_teta1=0,prev_teta2=0,prev_teta3=0,prev_teta4=0;
float tet1,tet2,tet3,tet4;

float teta1,teta2,teta3,teta4;
float link1=100,link2=100,link3=170,tbr=229;
float a,b,c,d,e,f,g,alpha,beta,teta,beta1,gamma;
int flag=0;

void gripper_open(){
    gripper.write(0);
}
void gripper_close(){
    gripper.write(90);
```

```

}
void default_position(){
    upper_arm.write(def_dof1);
    forearm.attach(def_dof2);
    wrist.attach(def_dof3);
    roll.write(def_dof4);
    gripper.write(0);
}
void roll_over(){
    roll.write(40);
    delay(150);
    roll.write(130);
    delay(150);
}
void polarisasi(){
    teta1=(teta1*(max_dof1 - min_dof1)/180)+min_dof1;
    teta2=(teta2*(max_dof2 - min_dof2)/180)+min_dof2;
    teta3=(teta3*(max_dof3 - min_dof3)/180)+min_dof3;
}
void inverse_kinematic(float x, float y){
    prev_teta1=teta1;
    prev_teta2=teta2;
    prev_teta3=teta3;

    a=sqrt(((tbr-y)*(tbr-y))+(x*x));
    c=link3+y-tbr;

    b=sqrt((c*c)+(x*x));
    d=(link1*link1)+(link2*link2)-(b*b);
    e=2*link1*link2;
    beta=acos(d/e)*180/3.14;
    teta=atan(c/x)*180/3.14;
    beta1=(180-beta)/2;
    gamma=atan(x/c)*180/3.14;

    teta1=beta1+teta;
    teta2=beta;
    teta3=beta1+gamma;
    teta3=180-teta3;

```

```

    if(teta1<0){teta1=0;}
    if(teta2<0){teta2=0;}
    else if(teta2<90){teta2=90; Serial.print("!!");}
    if(teta3<0){teta3=0;}
    polarisasi(); //setelah didapatkan nilai kinematik, maka di konversi

}

void robot_movement(){
    sel1=teta1-prev_teta1;
    sel2=teta2-prev_teta2;
    sel3=teta3-prev_teta3;

    if(sel1<0){sel1=sel1*(-1);}
    if(sel2<0){sel2=sel2*(-1);}
    if(sel3<0){sel3=sel3*(-1);}
    //if(sel4<0){sel4=sel4*(-1);}
    min_s=min(sel1,sel2);
    min_s=min(min_s,sel3);
    //min_s=min(min_s,sel4);
    if(min_s==0){min_s=1;}
    for(i=0;i<=min_s;i++){
        tet1=prev_teta1+(((teta1-prev_teta1)/min_s)*i);
        tet2=prev_teta2+(((teta2-prev_teta2)/min_s)*i);
        tet3=prev_teta3+(((teta3-prev_teta3)/min_s)*i);
        //tet4=prev_teta4+(((teta4-prev_teta4)/min_s)*i);
        //base_arm.write(tet1);
        upper_arm.write(tet1);
        forearm.write(tet2);
        wrist.write(tet3);
        delay(speed_delay);
    }
}

void setup(){
    upper_arm.attach(s1);
    forearm.attach(s2);
    wrist.attach(s3);
    roll.attach(s4);
    gripper.attach(s5);
}

```

```

Serial.begin(115200);
tes.begin(115200);
delay(500);
gripper_open();
upper_arm.write(90);
forearm.write(120);
}

void loop(){
  while(flag==0){
    if(tes.available()>0){
      inChar = tes.read();
    }

    delay(100);
    if(inChar==49){
      delay(1000);

      while(flag==0){
        roll.write(65);
        inverse_kinematic(170,110);

        robot_movement();
        delay(2000);
        gripper_close();
        delay(2000);
        roll.write(65);
        inverse_kinematic(170,160);

        robot_movement();
        delay(2000);

        tes.println(1);
        delay(100);
        flag=1;
      }
    }
  }
  while(flag==1){

```



```

if(tes.available()>0){
    inChar = tes.read();
}
delay(100);
if(inChar==50){
    delay(1000);
    while(flag==1){
        roll.write(65);
        inverse_kinematic(170,110);
        robot_movement();
        delay(2000);
        gripper_close();
        delay(2000);
        for(i=0;i<20;i++){
            roll_over();
        }
        delay(1000);
        roll.write(65);
        inverse_kinematic(170,160);
        robot_movement();
        delay(2000);
        tes.println(2);
        delay(100);
        flag=2;
    }
}
}
while(flag==2){
    if(tes.available()>0){
        inChar = tes.read();
    }
    delay(100);
    if(inChar==51){
        delay(1000);
        while(flag==2){
            roll.write(65);
            inverse_kinematic(170,110);
            teta1=teta1-25;
            teta2=teta2-0;

```

```

    teta3=teta3+15;
    robot_movement();
    delay(2000);
    gripper_close();
    delay(2000);
    for(i=0;i<210;i++){
        roll_over();
    }
    delay(1000);
    roll.write(65);
    inverse_kinematic(170,160);
    teta1=teta1-45;

    robot_movement();
    delay(2000);
    tes.println(3);
    delay(100);
    flag=3;
}
}
}
while(flag==3){
    if(tes.available(>0){
        inChar = tes.read();
    }
    delay(100);
    if(inChar==52){
        delay(1000);
        while(flag==3){
            roll.write(65);
            inverse_kinematic(170,110);

            robot_movement();
            delay(2000);
            gripper_close();
            delay(2000);
            for(i=0;i<20;i++){
                roll_over();
            }

```

```

        delay(1000);
        roll.write(65);
        inverse_kinematic(170,160);
        robot_movement();
        delay(2000);
        tes.println(4);
        delay(100);
        flag=4;
    }
}
}

while(flag==4){
    if(tes.available()>0){
        inChar = tes.read();
    }
    delay(100);
    if(inChar==53){
        delay(500);
        while(flag==4){
            roll.write(65);
            inverse_kinematic(170,110);
            robot_movement();
            delay(1000);
            gripper_open();
            delay(500);
            roll.write(65);
            inverse_kinematic(170,160);

            robot_movement();
            delay(500);
            tes.println(0);
            flag=5;

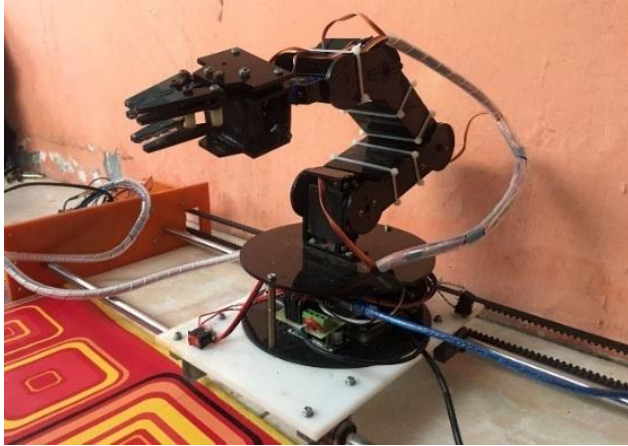
        }
    }
}
}
}

```

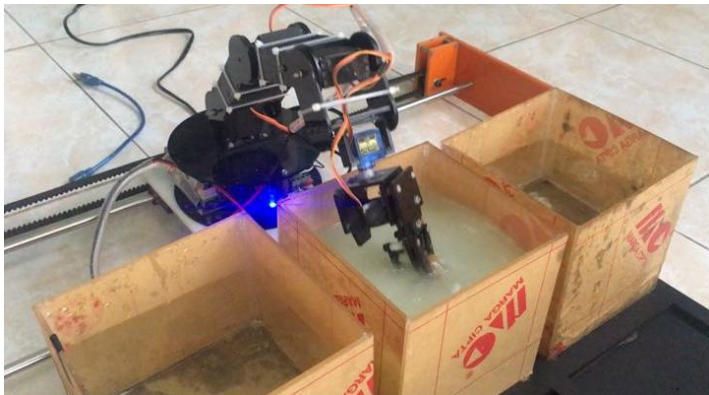
-----Halaman ini sengaja dikosongkan-----

## **LAMPIRAN B**

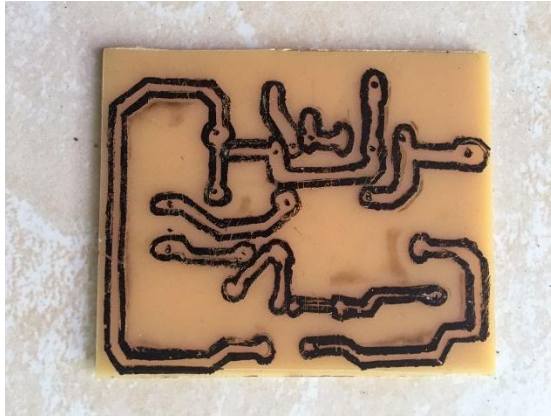
### **B.1 Dokumentasi**



**Gambar Keseluruhan Alat**



**Gambar Pengujian Keseluruhan Sistem (Menunjukkan pengambilan data keseluruhan alat pada bab 4.8)**



**Gambar Hasil Proses Etching Menggunakan Lengan Robot**

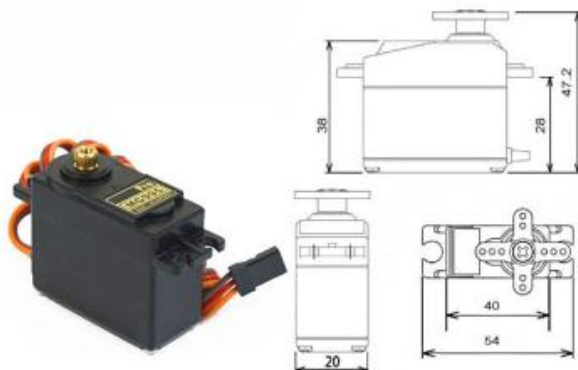


**Gambar Cairan  $HCL + H_2O_2 + H_2O$  yang digunakan untuk etching PCB**

## LAMPIRAN C

### C.1 Datasheet Motor Servo MG995

#### **MG995** High Speed Metal Gear Dual Ball Bearing Servo




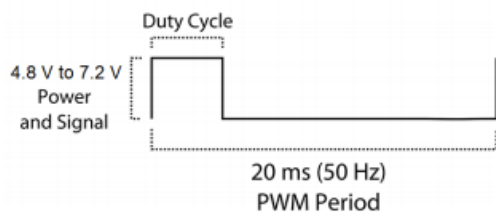
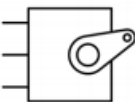
The unit comes complete with 30cm wire and 3 pin 'S' type female header connector that fits most receivers, including Futaba, JR, GWS, Cirrus, Blue Bird, Blue Arrow, Corona, Berg, Spektrum and Hitec.

This high-speed standard servo can rotate approximately 120 degrees (60 in each direction). You can use any servo code, hardware or library to control these servos, so it's great for beginners who want to make stuff move without building a motor controller with feedback & gear box, especially since it will fit in small places. The MG995 Metal Gear Servo also comes with a selection of arms and hardware to get you set up nice and fast!

#### Specifications

- Weight: 55 g
- Dimension: 40.7 x 19.7 x 42.9 mm approx.
- Stall torque: 8.5 kgf-cm (4.8 V ), 10 kgf-cm (6 V)
- Operating speed: 0.2 s/60° (4.8 V), 0.16 s/60° (6 V)
- Operating voltage: 4.8 V a 7.2 V
- Dead band width: 5  $\mu$ s
- Stable and shock proof double ball bearing design
- Temperature range: 0 °C – 55 °C

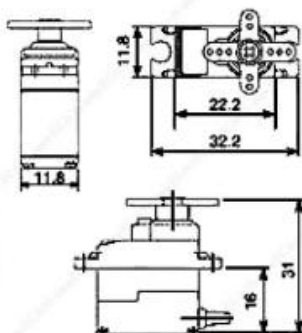
PWM=Orange ( )  
Vcc = Red ( + )  
Ground=Brown ( - )





## C.2 Datasheet Motor Servo MG90

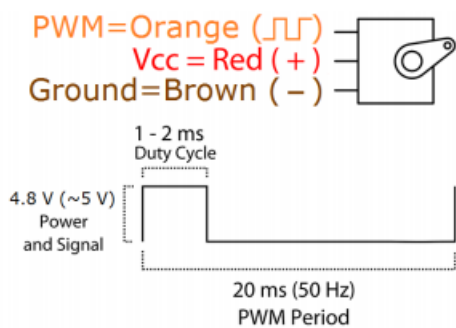
### SG90 9 g Micro Servo



Tiny and lightweight with high output power. Servo can rotate approximately 180 degrees (90 in each direction), and works just like the standard kinds but *smaller*. You can use any servo code, hardware or library to control these servos. Good for beginners who want to make stuff move without building a motor controller with feedback & gear box, especially since it will fit in small places. It comes with a 3 horns (arms) and hardware.

#### Specifications

- Weight: 9 g
- Dimension: 22.2 x 11.8 x 31 mm approx.
- Stall torque: 1.8 kgf-cm
- Operating speed: 0.1 s/60 degree
- Operating voltage: 4.8 V (~5V)
- Dead band width: 10  $\mu$ s
- Temperature range: 0 °C – 55 °C



Position "0" (1.5 ms pulse) is middle, "90" (~2 ms pulse) is all the way to the right, "-90" (~1 ms pulse) is all the way to the left.

### C.3 Datasheet Buck DC To DC Converter XL4005

**XLSEMI**

Datasheet

5A 300KHz 32V Buck DC to DC Converter

XL4005

#### Features

- Wide 5V to 32V Input Voltage Range
- Output Adjustable from 0.8V to 30V
- Maximum Duty Cycle 100%
- Minimum Drop Out 0.6V
- Fixed 300KHz Switching Frequency
- 5A Constant Output Current Capability
- Internal Optimize Power MOSFET
- High efficiency
- Excellent line and load regulation
- TTL shutdown capability
- EN pin with hysteresis function
- Built in thermal shutdown function
- Built in current limit function
- Built in output short protection function
- Available in TO-263 package

#### Applications

- LCD Monitor and LCD TV
- Digital Photo Frame
- Set-up Box
- ADSL Modem
- Telecom / Networking Equipment

#### General Description

The XL4005 is a 300KHz fixed frequency PWM buck (step-down) DC/DC converter, capable of driving a 5A load with high efficiency, low ripple and excellent line and load regulation. Requiring a minimum number of external components, the regulator is simple to use and include internal frequency compensation and a fixed-frequency oscillator.

The PWM control circuit is able to adjust the duty ratio linearly from 0 to 100%. An enable function, an over current protection function is built inside. When short protection function happens, the operation frequency will be reduced from 300KHz to 60KHz. An internal compensation block is built in to minimize external component count.



TO263-5L

Figure1. Package Type of XL4005

## Pin Configurations

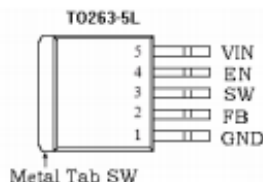


Figure2. Pin Configuration of XL4005 (Top View)

Table 1 Pin Description

Pin Number	Pin Name	Description
1	GND	Ground Pin. Care must be taken in layout. This pin should be placed outside of the Schottky Diode to output capacitor ground path to prevent switching current spikes from inducing voltage noise into XL4005.
2	FB	Feedback Pin (FB). Through an external resistor divider network, FB senses the output voltage and regulates it. The feedback threshold voltage is 0.8V.
3	SW	Power Switch Output Pin (SW). SW is the switch node that supplies power to the output.
4	EN	Enable Pin. Drive EN pin high to turn on the device, drive it low to turn it off.
5	VIN	Supply Voltage Input Pin. XL4005 operates from a 5V to 32V DC voltage. Bypass Vin to GND with a suitably large capacitor to eliminate noise on the input.

## Function Block

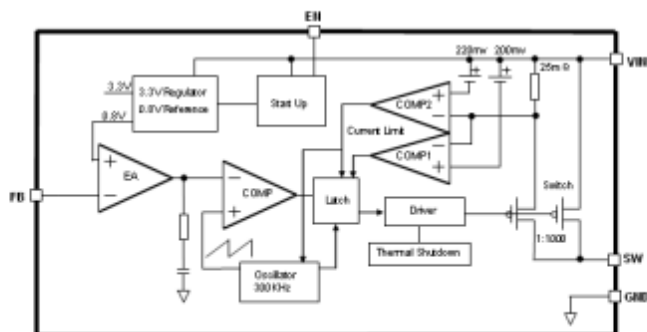


Figure3. Function Block Diagram of XL4005

## Typical Application Circuit

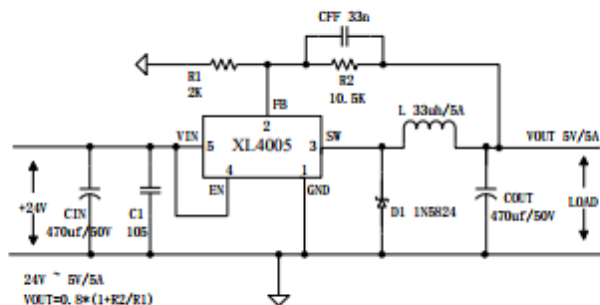


Figure4. XL4005 Typical Application Circuit 24V~5V/5A

5A 300KHz 32V Buck DC to DC Converter

XL4005

## Ordering Information

Order Information	Marking ID	Package Type	Packing Type Supplied As
XL4005E1	XL4005E1	TO263-5L	800 Units on Tape & Reel

XLSEMI Pb-free products, as designated with "E1" suffix in the par number, are RoHS compliant.

## Absolute Maximum Ratings (Note1)

Parameter	Symbol	Value	Unit
Input Voltage	$V_{in}$	-0.3 to 35	V
Feedback Pin Voltage	$V_{FB}$	-0.3 to $V_{in}$	V
EN Pin Voltage	$V_{EN}$	-0.3 to $V_{in}$	V
Output Switch Pin Voltage	$V_{Output}$	-0.3 to $V_{in}$	V
Power Dissipation	$P_D$	Internally limited	mW
Thermal Resistance (TO263) (Junction to Ambient, No Heatsink, Free Air)	$R_{JA}$	30	°C/W
Operating Junction Temperature	$T_J$	-40 to 125	°C
Storage Temperature	$T_{STG}$	-65 to 150	°C
Lead Temperature (Soldering, 10 sec)	$T_{LEAD}$	260	°C
ESD (HBM)		2000	V

**Note1:** Stresses greater than those listed under Maximum Ratings may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions above those indicated in the operation is not implied. Exposure to absolute maximum rating conditions for extended periods may affect reliability.

## 5A 300KHz 32V Buck DC to DC Converter

XL4005

### XL4005 Electrical Characteristics

 $T_A = 25^{\circ}\text{C}$ ; unless otherwise specified.

Symbol	Parameter	Test Condition	Min.	Typ.	Max.	Unit
<i>System parameters test circuit figure4</i>						
VFB	Feedback Voltage	$V_{in} = 5V \text{ to } 32V, V_{out} = 5V$ $I_{load} = 0.5A \text{ to } 5A$	0.776	0.8	0.824	V
Efficiency	$\eta$	$V_{in} = 12V, V_{out} = 5V$ $I_{out} = 5A$	-	90	-	%

### Electrical Characteristics (DC Parameters)

 $V_{in} = 12V, GND = 0V$ ,  $V_{in}$  &  $GND$  parallel connect a  $220\mu F/50V$  capacitor;  $I_{out} = 500mA$ ,  $T_A = 25^{\circ}\text{C}$ ; the others floating unless otherwise specified.

Parameters	Symbol	Test Condition	Min.	Typ.	Max.	Unit
Input operation voltage	$V_{in}$		5		32	V
Shutdown Supply Current	$I_{STBY}$	$V_{EN} = 0V$		60	200	$\mu A$
Quiescent Supply Current	$I_q$	$V_{EN} = 2V,$ $V_{FB} = V_{in}$		3	5	mA
Oscillator Frequency	$F_{osc}$		240	300	360	KHz
Switch Current Limit	$I_L$	$V_{FB} = 0$		8		A
EN Pin Threshold	$V_{EN}$	High (Regulator ON) Low (Regulator OFF)		1.4 0.8		V
EN Pin Input Leakage Current	$I_{IH}$	$V_{EN} = 2V$ (ON)		1	15	$\mu A$
	$I_{IL}$	$V_{EN} = 0V$ (OFF)		1	15	$\mu A$
Max. Duty Cycle	$D_{MAX}$	$V_{in} = 0V$		100		%

## Test Circuit and Layout guidelines

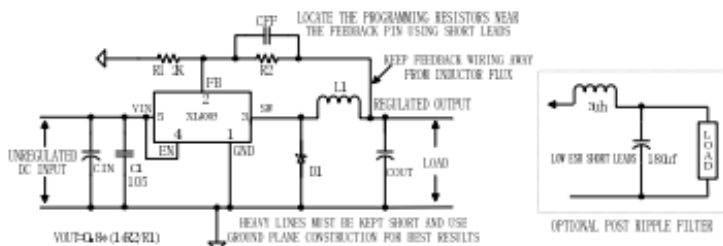


Figure5. Standard Test Circuits and Layout Guides

Select R1 to be approximately 2K, use a 1% resistor for best stability.

C1 and CFF are optional; in order to increase stability and reduce the input power line noise, CIN and C1 must be placed near to VIN and GND;

For output voltages greater than approximately 10V, an additional capacitor CFF is required. The compensation capacitor is typically between 100 pf and 33 nF, and is wired in parallel with the output voltage setting resistor, R2. It provides additional stability for high output voltage, low input-output voltages, and/or very low ESR output capacitors, such as solid tantalum capacitors.

$CFF = 1 / (31 * 1000 * R2)$ ; This capacitor type can be ceramic, plastic, silver mica, etc. (Because of the unstable characteristics of ceramic capacitors made with Z5U material, they are not recommended.)



Schottky Diode Selection Table

Current	Surface Mount	Through Hole	VR (The same as system maximum input voltage)				
			20V	30V	40V	50V	60V
1A		✓	1N5817	1N5818	1N5819		
3A		✓	1N5820	1N5821	1N5822		
		✓	MBR320	MBR330	MBR340	MBR350	MBR360
	✓		SK32	SK33	SK34	SK35	SK36
	✓			30WQ03	30WQ04	30WQ05	
		✓		31DQ03	31DQ04	31DQ05	
		✓	SR302	SR303	SR304	SR305	SR306
5A		✓	1N5823	1N5824	1N5825		
		✓	SR502	SR503	SR504	SR505	SR506
		✓	SB520	SB530	SB540	SB550	SB560
	✓			50WQ03	50WQ04	50WQ05	

## Typical System Application for 24V ~ 12V/4A Version

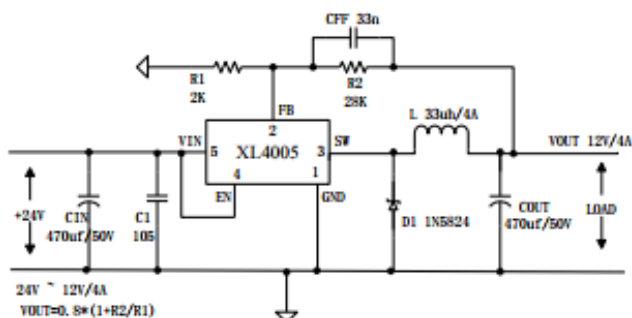


Figure6. XL4005 System Parameters Test Circuit (24V ~ 12V/4A)

## Typical System Application for 24V ~ 5V/5A

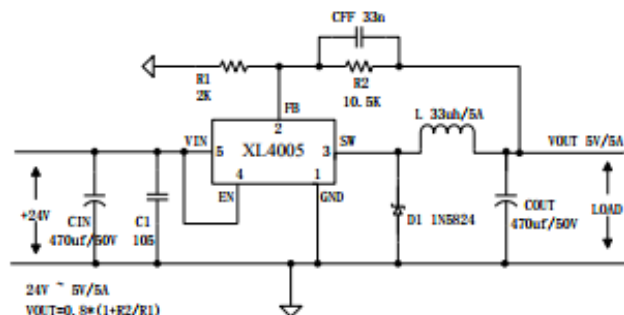
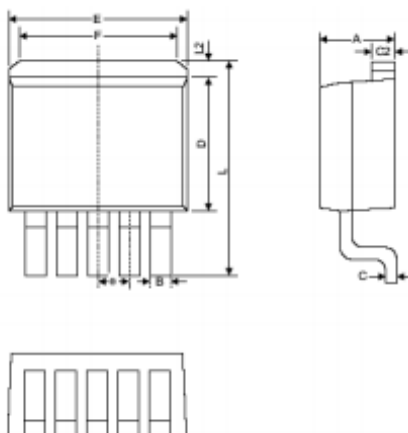


Figure7. XL4005 System Parameters Test Circuit (24V ~ 5V/5A)

**Package Information**

TO263-5L



Symbol	Dimensions in Millimeters		Dimensions in Inches	
	Min.	Max.	Min.	Max.
A	4.06	4.83	0.160	0.190
B	0.76	1.02	0.030	0.040
C	0.36	0.64	0.014	0.025
C2	1.14	1.40	0.045	0.055
D	8.64	9.65	0.340	0.380
E	9.78	10.54	0.385	0.415
e	1.57	1.85	0.062	0.073
F	6.60	7.11	0.260	0.280
L	15.11	15.37	0.595	0.605
L2	-	1.40	-	0.055

-----Halaman ini sengaja dikosongkan-----

## LAMPIRAN D

### DAFTAR RIWAYAT HIDUP



Nama : Gita Rizka Anandya  
TTL : Mojokerto, 12 Desember 1996  
Jenis Kelamin : Perempuan  
Agama : Islam  
Alamat : Perumahan Semolowaru Indah 2 Blok S/15  
Telp/HP : 089 678 400 935  
E-mail : anandyagita@gmail.com

### RIWAYAT PENDIDIKAN

1. 2002-2008 : SD N Puloniti Bangsal
2. 2008-2011 : SMP Negeri 1 Kota Mojokerto
3. 2011-2014 : SMA Negeri 2 Kota Mojokerto
4. 2014-2017 : D3 Teknik Elektro, Departemen Teknik Elektro Otomasi, Fakultas Vokasi Institut Teknologi Sepuluh Nopember

### PENGALAMAN KERJA

1. Kerja Praktek di PT Telkom Akses Witel Jatim Timur Tengah Sidoarjo

### PENGALAMAN ORGANISASI

1. Panitia Tendaku (Teknologi dan Edukasi untuk Rakyat) BEM FTI ITS 2015

-----Halaman ini sengaja dikosongkan-----